IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT :
APPLICATION No. : Group Art Unit :
FILING DATE : Examiner :
TITLE :

Hon. Commissioner of Patents and Trademarks,
Washington, D.C. 20231

SIR:

CERTIFIED TRANSLATION

I, Mio Hashimoto, am an official translator of the Japanese language into the English language and I hereby certify that the attached comprises an accurate translation into English of Japanese Application No. 11-039218, filed on February 17, 1999.

I hereby declare that all statements made herein of my own knowledge aretrue and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements andthe like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardizethe validity of the application or any patent issued thereon.

September 25, 2000
Date

Mio Hashimoto
Mio Hashimoto

BY EXPRESS MAIL NO. EL254113385US
Attorney Docket No. KOIK-T0185

09/647378

430 Rec'd PCT/PTO 28 SEP 2000

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re International Application of

International Serial No. PCT/JP00/00904
International filing date: 17 February 2000
For: Information Processing Apparatus And Method, And Program Storage Medium

VERIFICATION OF TRANSLATION

Honorable Commissioner of Patents and Trademarks
Washington, D.C. 20231
Sir:

Eiichi Tamura, a member of A.KOIKE & CO., of 11-Mori Bldg., 6-4, Toranomon 2-chome, Minato-ku, Tokyo 105-0001, Japan, declares:

(1) that he knows well both the Japanese and English languages;

(2) that he translated the above-identified International Application from Japanese to English;

(3) that the attached English translation is a true and correct translation of the above-identified International application to the best of his knowledge and belief; and

(4) that all statements made of his own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements are made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under 18USC 1001, and that such false statements may jeopardize the validity of the application or any patent issuing thereon.

September 21. 2000
Date

Eiichi Tamura

Patent Office

Japanese Government

This is to certify that the annexed is a true copy of the following application as filed with this Office.

Date of Application: February 17, 1999

Application Number: Patent Application
Ser. No.11-039218

Applicant: Sony Corporation

December 17, 1999

Commissioner,

Patent Office Takahiko kondo

[Document Name] Patent Application

[Reference Number] 9900113903

[Filing Date] February 17, 1999

[To] Hon.Commissioner,Patent Office

[IPC] G06F 19/00

[Inventor]

[Address] c/o Sony Corporation

7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo, Japan

[Name] Itaru Kawakami

[Inventor]

[Address] c/o Sony Corporation

7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo, Japan

[Name] Ryuji Ishiguro

[Inventor]

[Address] c/o Sony Corporation

7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo, Japan

[Name] Mitsuru Tanabe

[Inventor]

[Address] c/o Sony Corporation

7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo, Japan

[Name] Yuichi Ezura

[Patent Applicant]

[Identification Number] 000002185

[Name] Sony Corporation

[Representative] Nobuyuki Idei

[Patent Attorney]

[Identification Number] 100082131

[Patent Attorney]

[Name] Yoshio Inamoto

[Telephone Number] 03-3369-6479

[Indication of Charge]

[Number of Prepaid Ledger] 032089

[Amount] 21,000 yen

[List of Document]

[Document]	Specification	1
[Document]	Drawing	1
[Document]	Summary	1
[General Power of Attorney Number]		9708842

[Need of Proof] Yes

[Name of Document] SPECIFICATION

[Title of the Invention]

Information Processing Apparatus and Method and Furnishing Medium

[Claims]

[Claim 1]

An information processing apparatus comprising:

means for storing data;

a controlling means having a software which controls storage or read of the data into or from the data storage means; and

means provided in a hardware independent of the controlling means to decrypt and execute an encrypted program supplied from the controlling means and supply the result of the program execution to the controlling means;

the controlling means controlling the data storage or read to or from the data storage means based on the program execution result supplied from the program executing means.

[Claim 2]

The apparatus as set forth in Claim 1, wherein:

the data storage means stores also management information with which the data stored in itself is managed; and

the controlling means makes the program executing means execute a predetermined computation based on the management information.

[Claim 3]

The apparatus as set forth in Claim 1, wherein:

the controlling means is a CPU;

the data storing means is a hard disc; and

the program executing means is a CPU incorporated in a semiconductor IC other than a one in which the CPU as the controlling means is built.

[Claim 4]

An information processing method for use in an information processing apparatus comprising:

means for storing data;

a controlling means having a software which controls storage or read of the data into or from the data storage means; and

means provided in a hardware independent of the controlling means to decrypt and execute an encrypted program supplied from the controlling means and supply the result of the program execution to the controlling means;

the method comprising a step of:

controlling storage or read of data into or from the data storage means based on the result of the program execution by a program executing means.

[Claim 5]

A furnishing medium for use in an information processing apparatus comprising:

means for storing data;

a controlling means having a software which controls storage or read of the data into or from the data storage means; and

means provided in a hardware independent of the controlling means to decrypt and execute an encrypted program supplied from the controlling means and supply the result of the program execution to the controlling means;

the furnishing medium being intended for use in the controlling means and furnishing a computer-readable program comprising a step of controlling storage or read of the data into or from the data storing means based on a result of a program execution by the program executing means.

[Detailed Description of the Invention]

[0001]

[Technical Field of the Invention]

The present invention relates to an information processing apparatus and method, and to a furnishing medium, and more specifically, to an information processing apparatus and method adapted to prevent falsification of a software used with data in order to inhibit fraudulent copying of the data, and a furnishing medium furnishing an information processing program for the prevention of the fraudulent copying.

[0002]

[Prior Art]

Recently, with coming into extensive use of the digital technology, variable data, such as music data or picture data, can be digitally recorded or reproduced. The result is that data can be obtained which is not deteriorated in picture quality or sound quality even if the data is copied a number of times.

[0003]

[Problems to be Solved by the Invention]

However, with the progress in the digital technique, the following problems arise.

[0004]

(1) If, for example, digital music data is to be copied from a compact disc (CD) to a personal computer, music data from the CD is recorded on a hard disc directly or in an encoded form, so that a large number of duplications can be illicitly distributed over a network, such as Internet.

[0005]

(2) If digital music data is to be copied from the CD to a hard disc of the personal computer, there is no limitation on the number of times of copying, so that a large number of duplications are distributed.

[0006]

(3) If digital music data stored in a hard disc of a personal computer is to be transferred to an external equipment, such as memory stick walkman, the original digital music data is left in the hard disc after the data transfer, so that there persists

the risk of distribution of a large number of duplications.

[0007]

(4) In order to evade the problem (3), described above, the personal computer software may be formulated so that data of the hard disc as the data source will be erased, that is so that music data will be moved, after transferring the digital music data to the external equipment. However, if the contents of the hard disc are kept as backup on a different recording medium before moving the data and the backup data thus kept is re-stored in the hard disc after moving the data, the data which has been moved is left in the hard disc.

[0008]

(5) If the personal computer transfers digital music data in the hard disc to an external equipment such as a memory stick walkman, there is a risk that, since it is not confirmed to which equipment the data is transferred, digital music data be delivered to an unauthorized equipment.

[0009]

(6) If digital music data is to be delivered from an external equipment, such as a memory stick walkman, there is a risk that, since it is not confirmed which software is supervising the personal computer, the digital music data be delivered to an illicit software.

[0010]

(7) When music data reproduced from the CD is to be handled by the personal

computer, the International Standard Recording Code (ISRC) contained in the music number data can be used to check whether or not the plural numbers are the same number. However, in certain CDs, ISRC data is not contained, in which case it is impossible to verify whether or not the plural numbers are the same number.

[0011]

(8) The above-described functions are implemented on the personal computer under control by the software, so that, if the software is altered with inappropriate intention, it is probable that such an operation carried out is not the operation intended by a system designer.

[0012]

It is therefore an object of the present invention to prevent falsification of a software used with data in order to inhibit fraudulent copying of the data.

[0013]

[Means to Solve the Problem]

In one aspect, an information processing apparatus according to Claim 1comprises means for storing data, a controlling means having a software which controls storage or read of the data into or from the data storage means, and means provided in a hardware independent of the controlling means to decrypt and execute an encrypted program supplied from the controlling means and supply the result of the program execution to the controlling means. The controlling means controls the data storage or read to or from the data storage means based on the program execution

result supplied from the program executing means.

[0014]

In another aspect, an information processing method according to Claim 4 comprises a step of controlling storage or read of data into or from the data storage means based on the result of the program execution by a program executing means.

[0015]

In still another aspect, a furnishing medium according to Claim 5 is intended for use in the controlling means and furnishes a computer-readable program comprising a step of controlling storage or read of the data into or from the data storing means based on a result of a program execution by the program executing means.

[0016]

In the above information processing apparatus according to Claim 1, the information method according to Claim 4 and the information furnishing medium according to Claim 5, the controlling means comprised of a software controls storage or read of data into or from the data storage means based on the result of the program execution by a program executing means incorporated in a hardware.

[0017]

[Preferred Embodiment of the Invention]

Fig.1 shows an illustrative structure of a network system embodying the present invention. A personal computer 1 includes a central processing unit (CPU) 12, for executing variable processing, a memory 13 for transiently storing variable programs

and data and a hard disc 15 for storing a large quantity of programs and data. A CD-ROM (read-only memory) drive 14 reads out programs and data stored on a CD-ROM loaded in position. An audio input/output interface 16, having an IEC (International Electrotechnical Commission) 60958 terminal 16a, executes the digital audio input/output or analog audio input/output interfacing processing. An Internet connection interface 11 executes the interfacing processing with respect to the Internet 4. An interface 17 executes the interfacing processing for an adapter 7 or a memory stick walkman 6 and the interfacing processing with respect to an input unit 2 and a display 3.

[0018]

A CPU 32 of the adapter 7, as a monolithic semiconductor chip IC, loaded on the personal computer 1, cooperates with a CPU 12 of the personal computer 1 via an interface 31 to execute variable processing operations. A RAM 33 stores data or programs necessary for the CPU 32 to execute variable processing operations. A non-volatile memory 34 stores data that needs to be kept even after the power source of the personal computer 1 is turned off. In a ROM 36 is stored a program for decoding an encrypted program transferred from the personal computer 1. A real-time clock (RTC) 35 executes the timer operation to furnish the time information.

[0019]

The memory stick walkman 6 includes a non-volatile memory 23, in which to store digital music data furnished from the personal computer 1 via an interface 21,

and an authentication device 22. The authentication device 22 executes reciprocal authentication processing when exchanging data between the personal computer 1 and the non-volatile memory 23. The interface 21 executes interfacing processing with respect to the personal computer 1 and an interfacing processing of reading out music data stored in the non-volatile memory 23 and furnishing the read-out music data via e.g., a headphone to a user.

[0020]

The personal computer 1 is connected over the Internet 4 to an EMD (Electrical Music Distribution) server 5 to have music data furnished from the EMD server 5.

[0021]

Referring to the flowchart of Fig.2, the processing of transferring music data reproduced from a CD loaded on a CD-ROM drive 14 to the hard disc 15 for copying is explained. If the user actuates an input unit 2 to enter via interface 17 a command for the CPU 12 of transferring music data reproduced from a CD, not shown, loaded on the CD-ROM drive 14, to the hard disc 15 for copying, the CPU 12 at step S11 causes the GUI (graphical user interface) to be demonstrated on a display 3 for selecting a music number for copying.

[0022]

Specifically, the CPU 12 causes the TOC (table-of-contents) of the CD loaded on the CD-ROM drive 14 to acquire the information on the music number contained in the CD to display the information on the display 3. Alternatively, the CPU 12 reads

out ISRC (International Standard Recording Code) for each music number contained in the CD in order to acquire the information on the music number and in order to display the acquired in on the display 3. Still alternatively, the CPU 12 accesses an external database over the Internet 4 to acquire the information of the music number for displaying the corresponding GUI on the display 3. The user actuates the input unit 2, using the GUI of the display 3, to select the music number for copying.

[0023]

At step S12, the CPU 12 checks a terminal database stored in the hard disc 15. The terminal database checking processing is shown in detail in the flowchart of Fig.3.

[0024]

At step S31, the CPU 12 cooperates with the CPU 32 of the adapter 7 to compute hash values of the entire terminal database. At step S32, the CPU 12 compares the calculated values to the hash value stored last time.

[0025]

That is, there is formed a terminal database in the hard disc 15. In this terminal database, the ISRC numbers and the date and time of the copying of previously recorded music numbers are recorded in association with each other as the information usable for supervising the music data recorded on the hard disc 15. In the present case, the ISRC and the date and time of the copying of each of the three items 1 to 3 are recorded. The hash values of the entire terminal database, based on the ISRC numbers and the date and time of copying of the entire numbers recorded in the

terminal database are computed at step S38 by the CPU 32 of the adapter 7 at step S38 and recorded in the non-volatile memory 34. The hash values are obtained by applying a hash function to the data. The hash function in general is a unidirectional function for mapping data of variable lengths to values of short fixed length and has properties that collision between hash values is less likely to take place. Examples of the hash functions include SHA and MD5. The CPU 12 at step S31 computes hash values in the same way as the CPU 32. The CPU 12 at step S32 requests the CPU 32 to read out the hash values stored in the non-volatile memory 34 to compare the transferred hash values with the hash values the CPU 12 has just calculated.

[0026]

At step S33, the CPU 12 compares the hash values just calculated at step S31 with the last hash values of the terminal database stored in the non-volatile memory 34 as to possible coincidence. If the hash values are not coincident, the CPU 12 verifies that the terminal database has been altered. Thus, the CPU 12 at step S34 generates a message reading: "since the terminal database has been altered, copying is not possible" to output the message via the interface 17 to the display 3 for display thereon. The CPU 12 then terminates the processing. That is, in this case, the processing of reproducing music data recorded on the CD for copying on the hard disc 15 is prohibited.

[0027]

If the hash value computed at step S31 coincides with the previous hash value,

the program moves to step S35, where the CPU 12 acquires the ISRC number of the musical number, selected as musical number for copying, specified at step S11 (selected music number), from the CD. If no ISRC number is recorded on the CD, the CPU 12 reads out the TOC data of the CD and applies the hash function to the data to acquire data of a suitable length, such as 58 bits, to use the data in place of the ISRC number.

[0028]

At step S36, the CPU 12 verifies whether or not the ISRC number acquired at step S35 (that is the selected music number) is registered in the terminal database (Fig.4). If the ISRC number is not registered in the terminal database, the music number is not recorded as yet in the hard disc 15. Thus, the program moves to step S37, where the CPU 12 registers the ISRC number of the music number and the current time and date in the terminal database. Meanwhile, the CPU 12 exploits the output value of the RTC 35 of the adapter 7 transferred from the CPU 32. At step S38, the CPU 32 reads out data of the terminal database at the time point to transfer the read-out data to the CPU 32 of the adapter 7. The CPU 32 calculates the hash value of the transfer data to store the calculated hash value in the non-volatile memory 34. This hash value is compared at step S32 to the hash value stored last time.

[0029]

Then, at step S39, the CPU 12 sets an "unregistered" flag specifying that the selected number has not been registered in the terminal database. This flag is used at

the step S13 of Fig.2, as later explained, to check whether or not the selected music number has been registered in the terminal database.

[0030]

If, at step S36, the ISRC number of the selected music number has been verified to be registered in the terminal database, the selected music number is the music number registered at least once in the hard disc 15. Thus, in this case, the program moves to step S40 where the CPU 12 checks whether or not the current date and time (current date and time outputted by the RTC 35 of the adapter 7) is not less than 48 hours past the date and time of registration of the selected music number registered in the terminal database. If the current time is past the date and time of registration by not less than 48 hours, the music number has been registered at least once on the hard disc 15. However, since the time of 48 hours or more elapsed since the music number was registered, there is no particular harm if the music number is copied again. Thus, in this case, the copying on the hard disc 15 is allowed. Therefore, the program moves to step S41 where the CPU 12 changes the date and time of the terminal database from the past date and time of registration to the current date and time (date and time outputted by the RTC 35). The program then reverts to step S38 where the CPU 12 causes the CPU 32 to compute the hash value of the entire terminal database for storage in the non-volatile memory 34. The CPU 12 at step S39 sets an "unregistered" flag for the music number.

[0031]

On the other hand, if the current time is not past by not less than 48 hours as from the date and time of registration, the copying of the selected music number on the hard disc 15 is prohibited. In this case, the program moves to step S42 where the CPU 12 sets the "registered" flag for the selected music number.

[0032]

By the above-described terminal database checking, a flag specifying whether or not the selected music number has been registered in the hard disc 15 is set..

[0033]

Reverting to Fig.2, the CPU 12 verifies, from the above-mentioned flag, whether or not the selected music number has already been registered in the terminal database. If the selected music number has been registered, the program moves to step S14, where the CPU 12 causes the display 3 to demonstrate a message reading: "This music number has been in the copied state for less than 48 hours and so the music number cannot be copied". This allows the user to comprehend the reason why the music number may not be copied on the hard disc 15.

[0034]

If, at step S13, the selected music number has been verified not to be registered in the terminal database, the program moves to step S15, where the CPU 12 controls the CD-ROM drive 14 to read out the music data from the CD loaded therein. In this music data, a watermark code is inserted at a pre-set position, as shown in Fig.5. At step S16, the CPU 12 extracts the watermark code contained in the music data to

verify at step S17 whether or not the watermark code indicates copying inhibition. If the watermark code indicates copying inhibition, the program moves to step S18 where the CPU 12 causes a message reading: "copying in inhibited" to be demonstrated on the display 3 via the interface 17 to terminate the copying processing.

[0035]

If conversely it has been found at step S17 that the watermark is not indicating copying inhibition, the program moves to step S19 where the CPU 12 compresses music data by software processing in accordance with, for example, adaptive transform acoustic coding system(ATRAC) (registered trademark). At step S20, the CPU 12 encrypts the music data, using a pre-set cryptographic key stored in the memory 13, in accordance with an encrypting method, such as data encryption standard (DES) system or the fast encipherment algorithm (FEAL) system. The cryptographic key may also be a random number generated by software or a random number generated by the CPU 32 of the adapter 7. By performing the encryption processing not only by the personal computer 1 alone but also by cooperation between the personal computer 1 and the CPU 32 of the adapter 7 in executing the encryption processing, it is possible to realize encryption which renders deciphering more difficult.

[0036]

Then, at step S21, the CPU 12 causes the encrypted data to be transferred to the hard disc 15 to store the encrypted data as a file along with a filename. Alternatively, the position information of the filename, such as is given by the number of bytes from

the leading end, may be accorded as a part of the file for storage.

[0037]

This storage processing may be performed independently of or simultaneously with the above-mentioned encoding and encryption processing.

[0038]

At step S22, the CPU 12 encrypts the encrypted cryptographic key with which the music data is encrypted in accordance with an encrypting method, such as the DES system or the FEAL system, using the storage key stored in the pre-set memory 13 to store it in a music number database of the hard disc 15.

[0039]

At step S23, the CPU 12 groups together the information on the stored files, encrypted cryptographic key, information on the music number and the information on the name of the music number, inputted by the user via GUI, as a set, and registers these elements of the set in the music number database of the hard disc 15. At step S24, the CPU 32 causes the CPU 32 to compute the hash values of the entire music number database for storage in the non-volatile memory 34.

[0040]

In this manner, the music number database, shown for example in Fig.6, is registered on the hard disc 15. In the present case, the filenames of the items 1 to 3, encrypted cryptographic key, name of the music number, duration of the music number, playback conditions (date and time of start and end and limitation on the

number of times), a number of times of playback counter, playback time charging conditions, copying conditions (number of times), a number of times of copying counter and copying conditions (SCMS).

[0041]

Referring to the flowchart of Figs.7 to 9, the processing of moving music data from the hard disc 15 to the non-volatile memory 23 of the memory stick walkman 6 is explained. At step S51, the CPU 12 computes the hash values of the entire music number database. At step S52, the CPU 12 compares the computed hash values to the hash values computed under control by the CPU 32 and stored in the non-volatile memory 34. In case of non-coincidence between the two hash values, the CPU 12 advances to step S53 to cause a message: "it is feared that the music number database has ben altered with inappropriate intention" to be demonstrated on the display 3 to terminate the processing. The processing in this case is similar to that from steps S31 to S34 of Fig.3. In this case, the music data is not moved from the hard disc 15 to the memory stick walkman 6.

[0042]

Then, at step S54, the CPU 12 causes the information on the music number registered in the music number database formed in the hard disc 15 for demonstration on the display 3 as the GUI for selection. Based on this GUI for selection, the user selects the music number to be moved from the hard disc 15 to the memory stick walkman 6 by actuating the input unit 2. Then, at step S55, the CPU 12 checks the

playback conditions, copying conditions or the playback time charging conditions of the music number selected at step S54. This processing will be explained in detail with reference to the flowchart of Fig. 10.

[0043]

Then, at step S56, authentication processing is carried out reciprocally between the CPU 12 of the personal computer 1 and the authentication device 22 of the memory stick walkman 6, whereby the session key is co-owned.

[0044]

It is assumed that a master key K_M is pre-stored in a non-volatile memory 23 of the memory stick walkman 6, and that a personal key K_P and the ID are pre-stored in the memory 13 of the personal computer 1. The authentication device 22 is fed from the CPU 12 with the ID pre-stored in the memory 13, and generates a key which is the same as the personal key of the personal computer 1 stored in the memory, by applying the hash function to the ID and the master key K_M which is in its possession. By so doing, the common personal key is co-owned by the personal computer 1 and the memory stick walkman 6. Using this personal key, it is possible to generate a transient session key.

[0045]

Alternatively, the ID and the master key K_{MP} are pre-stored in the memory 13 of the personal computer 1, at the same time as the ID and the master key K_{MM} of the memory stick walkman 6 are stored in the non-volatile memory 23 of the memory stick

walkman 6. The respective IDs and master keys are reciprocally transmitted so that one of the personal computer 1 and the memory stick walkman 6 applies the hash function to the ID and the master key transmitted from the other to generate a personal key of the other. From this personal key, a transient session key is generated further.

[0046]

Meanwhile, the IOS (International Organization for Standardization) 9798-2, for example, may be utilized as the authentication method.

[0047]

If the reciprocal authentication has not been performed correctly, the processing comes to a close. If the reciprocal authentication has been performed correctly, the CPU 12 reads out the filename of the selected number from the music number database, while reading out music data of the filename, such as musical data encrypted by the processing of step S20 of Fig.2, from the hard disc 15. At step S58, the CPU 12 executes the processing of converting the encoding system of digital music data read out at step S57 (processing of step S19), encrypting system (processing of step S20) or the format into those of the memory stick walkman 6. This converting processing will be explained subsequently in detail with reference to the flowchart of Fig.12.

[0048]

At step S59, the CPU 12 encrypts the music data, converted at step S58, with the session key co-owned by the reciprocal authentication processing of step S56, to

transfer the encrypted music data to the memory stick walkman 6 via the interface 17. On reception at step S60 of the music data transmitted via interface 21, the authentication device 22 of the memory stick walkman 6 causes the music data to be directly stored in the non-volatile memory 23.

[0049]

At step S61, the CPU 12 converts the playback conditions (date and time of playback start and end, limitation on the number of times of playback etc) of the selected music number, registered in the music number database, into those of the form supervised by the memory stick walkman 6. At step S62, the CPU 12 converts the SCMS information in the copying condition registered in the music number database of the selected music number into that of the form supervised by the memory stick walkman 6. At step S63, the CPU 12 transfers the playback condition converted at step S61 and the SCMS information converted at step S62 to the memory stick walkman 6. The authentication device 22 stores the transferred playback condition and the SCMS information in the non-volatile memory 23.

[0050]

At step S64, the CPU 12 transfers the playback condition, playback time charging conditions or the copying conditions, registered in the music number database of the selected music number, in those of the form used by the CPU 12 in the music number database, for storage in the non-volatile memory 23.

[0051]

At step S65, the CPU 12 reads out the encrypted key of the selected music number from the music number database. At step S66, the CPU 12 decodes the cryptographic key with the storage key stored in the memory 13 and encrypts the decoded cryptographic key with the session key. The CPU 12 transfers the cryptographic key, encrypted with the session key, to the memory stick walkman 6.

[0052]

The authentication device 22 of the memory stick walkman 6 at step S67 decodes the cryptographic key, transferred from the personal computer 1, using the session key co-owned in the reciprocal authentication processing, and encrypts the decoded cryptographic key using a storage key which is in its possession. The CPU 12 causes the encrypted cryptographic key to be stored in the non-volatile memory 23 in association with the already stored data.

[0053]

On termination of the storage of the cryptographic key, the authentication device 22 at step S68 advises the personal computer 1 of the storage of the cryptographic key. On reception of this notice from the memory stick walkman 6, the CPU 12 of the personal computer 1 at step S69 deletes the file of the music data from the hard disc 15, while also deleting the set of the elements of the music number from the music number database. This realizes movement instead of copying. At step S70, the CPU 12 transfers the data of the music number database to the CPU 32 of the adapter 7 to cause the CPU 32 to compute the hash values of the entire database for

storage in the non-volatile memory 23. This hash value is used at the aforementioned step S52 as the hash value stored last time.

[0054]

The check processing of checking the playback condition of the selected music number at step S55 of Fig.7 is explained. At step S81, the CPU 12 causes various conditions to be read out from the music number database. The CPU 12 at step S82 verifies whether or not the number of times of copying, among the conditions read out at step S81, has already exceeded the limit number of times of copying. If the number of times of copying has already exceeded the limit number of times of copying, the copying is no longer allowed. Thus, the CPU 12 advances to step S83 where the CPU 12 causes a message reading, for example,: "the number of times of copying has already exceeded the limit number of times of copying", to be demonstrated on the display 3 to terminate the processing. If it is verified at step S82 that the number of times of copying has not exceeded the limit number of times of copying, the program moves to step S84 to check whether or not the current date and time is past the playback end date and time. The current date and time may be that outputted by the RTC 35 of the adapter 7. This prohibits the current date and time of the personal computer 1 from being intentionally corrected by the user to past values. The CPU 12 is fed from the CPU 32 with this current date and time to give the decision of step S84 on its own. The CPU 12 may also route at step S81 the playback condition read out from the music number database to the CPU 32 of the adapter 7 to cause the CPU 32

to execute the decision processing of step S84.

[0055]

If the current date and time is past the playback end date and time, the program moves to step S85, where the CPU 12 erases the selected music number from the hard disc 15, at the same time as it erases the information on the selected music number from the music number database. At step S86, the CPU 12 causes the CPU 32 to calculate the hash value of the music number database to store the calculated value in the non-volatile memory 34. The processing then comes to a close. Thus, in this case, the music data is not moved.

[0056]

If it is verified at step S84 that the current date and time is not past the playback end date and time, the program moves to step S87 where the CPU 12 verifies whether or not the playback time charging conditions for the selected music number (such as, for example, the fee per each reproduction), is registered in the music number database. If the playback time charging conditions are registered, the CPU 12 at step S88 communicates with the memory stick walkman 6 to check whether or not the memory stick walkman 6 has the charging function. If the memory stick walkman 6 does not have the charging function, the selected music number may not be transmitted to the memory stick walkman 6. Thus, the CPU 12 at step S89 causes a message reading, for example: "the destination of transfer does not have the charging function" to be demonstrated on the display 3 to terminate the moving processing for music data.

[0057]

If it has been found at step S87 that the playback time charging function is not registered, or if it has been found at step S88 that the memory stick walkman 6 has the charging function, the program moves to step S90, where the CPU 12 verifies whether or not the other playback functions, such as the limit number of times of reproduction, has been registered for the selected music number. If the other playback conditions are registered, the program moves to step S91 where the CPU checks whether or not the memory stick walkman 6 has the function of observing the playback functions. If the memory stick walkman 6 does not have the function of observing the playback functions, the program moves to step S92 where the CPU 12 demonstrates a message reading, for example: "the device of the destination of transfer does not have the function of observing the playback conditions" on the display 3 to terminate the processing.

[0058]

If it is verified at step S90 that the playback conditions are not registered, or it is verified at step S91 that the memory stick walkman 6 lacks in the function of observing the playback conditions, the processing of checking the playback conditions etc is terminated to revert to the step S56 of Fig.7.

[0059]

Fig.11 shows an example of the playback conditions supervised by the memory stick walkman 6, that is the playback conditions that can be observed by the memory

stick walkman 6. In the present example, the playback start date and time and playback end date and time are registered for each number of the items 1 to 3, however, the number of times of playback is registered only for the item 2, while it is not registered for the items 1 or 3. Thus, if the music number of the item 2 is selected, the playback conditions for the number of times of playback can be observed. However, if the music number of the item 1 or 3 is selected, the playback conditions for the number of times of playback cannot be observed.

[0060]

Referring to the flowchart of Fig.12, the format conversion processing at step S58 of Fig.7 is explained in detail. At step S101, the CPU 12 checks the format of the selected music number recorded on the hard disc 15 (playback conditions, using conditions or copying conditions). At step S102, the CPU 12 checks the condition that can be set on the counterpart equipment, here the memory stick walkman 6. That is, the CPU 12 inquires into the conditions that can be set on the authentication device 22 of the memory stick walkman 6 and acquires a response. The CPU 12 at step S103 gives a decision based on the format condition which is registered in the music number database and which can be set on the counterpart equipment as checked at step S102.

[0061]

At step S104, the CPU 12 decides whether or not there is any condition that can be set. If there is no condition that can be set, the program moves to step S105 where the CPU 12 inhibits the processing of moving the music data to the memory stick

walkman 6. That is, since the memory stick walkman 6 cannot observe the condition registered in the music number database, the memory stick walkman 6 is prohibited from moving the music data.

[0062]

If it is verified at step S104 that there is a condition that can be set at step S104, the CPU 12 advances to step S106 where the CPU 12 converts the condition to that of the functional format condition of the counterpart device. At step S107, the CPU 12 sets the converted condition on the counterpart equipment. The result is that the memory stick walkman 6 can reproduce the music data in accordance with the as-set conditions, that is so as to observe these conditions.

[0063]

Referring to the flowchart of Figs.13 to 15, the processing of copying music data from the hard disc 15 to the memory stick walkman 6 is explained. The processing from step S111 of Fig.13 to step S127 of Fig.15 is similar to the processing of step S51 to step S67 of Figs.7 to 9 of moving music data from the hard disc 15 to the memory stick walkman 6. That is, in this case, the music number database is checked as to possible falsification, after which the reproduction condition of the selected music number is checked. The reciprocal authentication processing between the memory stick walkman 6 and the personal computer 1 then is carried out, after which musical data is transferred from the hard disc 15 of the personal computer 1 to the non-volatile memory 23 of the memory stick walkman 6 for storage therein. Then,

at step S128, the CPU 12 of the personal computer 1 increments by 1 the number of times of copying counter of the music number database. At step S129, the CPU 12 causes the hash values of the entire music number database to be calculated and stored in the non-volatile memory 34.

[0064]

Referring to the flowchart of Fig.16, the processing of moving music data from the memory stick walkman 6 to the hard disc 15 is explained. At step S161, the CPU 12 of the personal computer 1 requests the authentication device 22 of the memory stick walkman 6 to read out the information of the music number stored in the non-volatile memory 23. The authentication device 22 is responsive to this request to transmit the information on the music number stored in the non-volatile memory 23 to the personal computer 1. Based on this information, the authentication device 22 causes the display 3 to demonstrate the GUI for selecting the music number stored in the non-volatile memory 23. The user actuates the input unit 2 to specify the music number to be moved from the memory stick walkman 6 to the hard disc 15, based on the GUI.

[0065]

At step S162, the CPU 12 executes the reciprocal authentication processing with respect to the authentication device 22 to co-own the session key. This processing is similar to that at step S56 of Fig.7.

[0066]

Then, at step S163, the authentication device 22 reads out the music data of the selected music number stored encrypted in the non-volatile memory 23 to transfer the read-out data to the personal computer 1. At step S164, the CPU 12 of the personal computer 1 accords a filename as a file to music data transferred from the memory stick walkman 6 to store the file in the hard disc 15. This storage can be done by according the position information of the filename, such as, for example, the number of bytes as counted from the leading end, as a part of a file.

[0067]

At step S165, the authentication device 22 reads out the encrypted cryptographic key of the selected music number stored in the non-volatile memory 23, decodes the key with its own storage key, encrypts the decoded key with the session key and transfers it to the personal computer 1. This cryptographic key has been stored in the non-volatile memory 23 by the processing at step S67 of Fig.9.

[0068]

If the cryptographic key from the memory stick walkman 6 is transferred from the memory stick walkman 6 to the CPU 12 of the personal computer 1, the CPU 12 decodes the key with the session key and encrypts it with its own storage key. At step S167, the CPU 12 registers the filename of the music data file stored at step S164, the name of the music data inputted via GUI by the user or the cryptographic key encrypted at step S166, in the music number database of the hard disc 15. At step S168, the CPU 12 causes the CPU 32 to compute the hash values of the entire

database to store the computed hash values in the non-volatile memory 34.

[0069]

At step S169, the CPU 12 of the personal computer 1 advises the memory stick walkman 6 of the effect of storage of the cryptographic key to make a request for deleting the music data of the music number. If the musical data of the music number is requested by the personal computer 1, the authentication device 22 at step S170 deletes the musical data of the music number stored in the non-volatile memory 23.

[0070]

The processing for copying the musical data from the memory stick walkman 6 to the hard disc 15 is explained with reference to the flowchart of Fig.17. The processing of steps S181 to S188 shown in Fig.17 is similar to the processing of steps S161 to S168 in the processing of moving musical data from the memory stick walkman 6 to the hard disc 15. That is, the copying processing is basically the same as that for moving except that the steps S169, S170 in Fig.16 are omitted. Therefore, the corresponding description is omitted for clarity.

[0071]

Referring to the flowchart of Fig.18, the processing of copying the musical data transferred from the EMD server 5 in the hard disc 15 is explained. If accessing to the EMD server 5 is commanded by the user via the input unit 2, the CPU 12 at step S201 controls the Internet connection interface 11 to permit accessing to the EMD server 5 via the Internet 4. The EMD server 5 is responsive to this accessing to transfer the

information held by it, such as the number or name of the music number or the variable information, via Internet 4 to the personal computer 1. On acquisition of the information via the Internet connection interface 11, the CPU 12 of the personal computer 1 demonstrates the information on the display 3 via the interface 17. The user exploits the GUI demonstrated on the display 3 to specify the music number desired to be copied. This specifying information is transferred via the Internet 4 to the EMD server 5. At step S203, the CPU 12 executes the reciprocal authentication processing with the EMD server 5 via the Internet 4 to co-own the session key.

[0072]

The reciprocal authentication processing between the personal computer 1 and the EMD server 5 may be carried out using a public key and a secret key prescribed in ISO 9798-3, as an example. In this case, the personal computer 1 owns its own secret key and the public key of the EMD server 5 from the outset, whilst the EMD server 5 has its own secret key, to enable the reciprocal authentication processing to be executed. The public key of the personal computer 1 may be transferred from the EMD server 5, or the certificate previously allocated to the personal computer 1 may be transferred from the personal computer 1 to the EMD server 5. In the latter case, the certificate may be confirmed by the EMD server 5 to produce the public key. Also, the CPU 12 at step S204 executes the charging processing with respect to the EMD server 5. The charging processing will be explained subsequently by referring to the flowchart of Fig.19.

[0073]

Then, at step S205, the EMD server 5 transfers the encrypted music data of the music number specified at step second light transmission/reception device 202 to the personal computer 1 via the Internet 4. At this time, the time information is also transmitted as necessary. At step S206, the CPU 12 accords the filename to the music data transferred thereto to store the data as a file. At step S207, the EMD server 5 encrypts the cryptographic key of the music number, using the session key co-owned with the personal computer 1 at step S203, to transfer the encrypted cryptographic key to the personal computer 1.

[0074]

At step S208, the CPU 12 decodes the cryptographic key, transferred from the EMD server 5, by itself or in cooperation with the CPU 32 of the adapter 7, to encrypt the decoded cryptographic key with its own storage key. At step S209, the CPU 12 groups together the filename of the music number, information on the music number, the information on the input name of the music number, and the encrypted cryptographic key, as a set, and registers the set in the music number database of the hard disc 15. At step S210, the CPU 12 causes the hash values of the entire music number database to be computed to store the computed hash values in the non-volatile memory 34.

[0075]

At step S205, the EMD server 5 transmits time data, along with the music data,

to the personal computer 1. This time data is transferred from the personal computer 1 to the adapter 7. On reception of the time data, transmitted from the personal computer 1, the CPU 32 of the adapter 7 at step S211 corrects the time of the RTC 35. Since the time information of the RTC 35 of the adapter 7 is corrected in this manner based on the time information obtained from an external equipment recognized to be authorized as a result of the reciprocal authentication, the correct time information can be kept at all times in the adapter 7.

[0076]

Referring to the flowchart of Fig.19, the processing concverning the charging at step S204 of Fig.18 is explained in detail. At step S221, the personal computer 1 reads out the price information of the selected music number, specified at step S201, to write the read-out information on a charging log on the hard disc 15. Fig.20 shows an example of such charging log. In this case, the user copies items 1 to 3 from the EMD server 5 , with the charges for the items 1 and 2 being 50 yen and with that of the item 3 being 60 yen. The hash value of the charging log at this time point is also computed by the CPU 32 and registered in the non-volatile memory 34.

[0077]

Then, at step S222, the CPU 12 of the personal computer 1 reads out the charging log written at step S221 from the hard disc 15 to transfer the read-out charging log via the Internet 4 to the EMD server 5. The EMD server 5 at step S223 executes the charging computational processing based on the charging log transferred

from the personal computer 1. That is, the EMD server 5 adds the charging log transferred by the user of the personal computer 1 to a database enclosed therein for updating. At step S224, the EMD server 5 decides whether or not the charging log should be settled at once. If it is decided that the settlement be carried out at once, the program moves to step S25 where the EMD server 5 transfers the name of an article necesssary forsettlement or amount to a settlement server, not shown. At step S226, the settlement server executes the settlement for the user of the personal computer 1. If it is decided at step S224 that the settlement not be carried out at once, the processing at steps S225 and S226 is skipped. That is, this processing is subsequently executed periodically, such as once a month.

[0078]

Referring to the flowchart of Figs.21 and 22, the processing of copying the playback music data from a CD player, supplied from an IEC 60958 terminal 16a of the audio input/output interface 16, not shown, to the hard disc 15, is explained. At step S241, the user connects the IEC60958 of the CD player to the IEC 60958 terminal 16a of the audio input/output interface 16 of the personal computer 1. At step S242, the user acts on the input unit 2 to enter the name of the music number to be copied from the CD player. At step S243, the user actuates a button of the music number to be copied from the CD player to start the reproduction of the CD player. If the CD player and the personal computer 1 are interconnected by a line for exchanging control signals, a playback start command is inputted via the input unit 2 of the personal

computer 1 to start the reproduction of the CD player.

[0079]

If the reproduction of a CD is started in the CD player, the music data outputted by the CD player is transferred via IEC 60958 terminal 16a to the personal computer 1. At step S245, the CPU 12 reads out SCMS (serial copy management system) data from input data at the IEC 60958 terminal 16a. In this SCMS data, there are contained the information on copying inhibition, copying allowed only once or copying free. The CPU 12 at step S246 checks whether or not the SCMS data indicates copying inhibition. If the result indicates copying inhibition, the program moves to step S247 where the CPU 12 causes the display to demonstrate a message reading: "copying is inhibited" to terminate the copying. That is, in such case, the copying on the hard disc 15 is inhibited.

[0080]

If the CPU 12 verifies at step S246 that the SCMS information read out at step S245 is not indicating copying inhibition, it advances to step S248 to read out the watermark code to check at step S249 whether or not the watermark code indicates the copying inhibition. If the watermark code indicates copying inhibition, the program moves to step S247 where a pre-set message is displayed as above to terminate the copying processing.

[0081]

If it is verified at step S249 that the watermark code is not indicating copying

inhibition, the program moves to step S250 to carry out terminal database check. If, as a result of the terminal database check, the selected music number has already been registered, the processing is terminated with the steps S251 and S252. This processing is similar to that of steps S13 and S14.

[0082]

If the selected music number has not been registered on the hard disc 15, the registration processing is executed at steps S253 to S258. Since the processing of steps S253 to S258 is similar to that from steps S19 to S24 of Fig.2, except that the SCMS information supplied from the IEC60958 terminal at step S257 is also registered in the music number database, the processing is not explained herein specifically.

[0083]

Referring to the flowchart of Figs.23 and 24, the processing of reproducing music data from the hard disc 15 to the IEC 60958 terminal 16a is explained. At steps S271 to S273, as at steps S111 to S113 of Fig.13, the hash values of the entire music number database are computed to verify whether or not these hash values coincide with those stored last time, by way of performing the check as to whether or not the music number database has been altered with inappropriate intention. If it is verified that the music number database has not been altered in this manner, the program moves to step S274 where the CPU 12 accesses the music number database of the hard disc 15 to read out the information of the music number registered therein to

demonstrate the read-out information on the display 3. The user, viewing the display, acts on the input unit 2 to select the music number to be reproduced. At step S275, the CPU 12 executes the check of the playback condition etc of the selected music number. The processing for checking the playback condition etc will be explained in detail by referring to the flowchart of Fig.25.

[0084]

Then, at step S276, the CPU 12 reads out the cryptographic key of the music number selected at step S274 to decode the read-out cryptographic key with the storage key. At step S277, the CPU 12 reads out the SCMS information of the selected music number from the music number database to decide the SCMS information outputted at the IEC60958 terminal in accordance with the rule of the SCMS system. For example, if limitations are placed on the number of times of reproduction, the number of times of reproduction is incremented by 1 to provide the new SCMS information. At step S278, the CPU 12 further reads out the ISRC of the selected music number from the music number database.

[0085]

Then, at step S279, the CPU 12 reads out the filename of the selected music number from the music number database to read out the music data from the hard disc 15 based on the read-out filename. The CPU 12 further reads out the cryptographic key corresponding to the music data from the music number database to decode the read-out key with a storage key. The CPU 12 then decodes the encrypted music data,

using the decoded cryptographic key. The CPU 12 then further decodes the encoded music data. At step S280, the CPU 12 outputs the digital music data decoded at step S279 at the IEC 60958 terminal 16a, in accordance with the prescriptions of IEC 60958, along with the SCMS information decided at step S277 and the ISRC information read out at step S278. The CPU 12 also analogizes the digital music data to output the analog music data at an analog output terminal of the audio input/output interface 16.

[0086]

At step S281, the CPU 12 increments the value of the number of times of playback counter in the music number database by 1. At step S282, it is checked whether or not the playback time charging condition has been added in the selected music number. If the playback time charging condition has been added, the program moves to step S283, where the CPU 12 writes the corresponding charges in the charging log.. At step S284, the CPU 12 causes the CPU 32 to compute the hash values of the entire music number database for storage in the non-volatile memory 34. If it has been verified at step S282 that the playback time charging condition has not been added to the selected music number, the processing at steps S283 and S284 is skipped.

[0087]

Referring to the flowchart of Fig.25, the processing of checking the playback conditions etc of the step S275 of Fig.23 is explained. At step S301, the CPU 12 reads

out the variable conditions of the music number database. At step S302, the CPU 12 checks whether or not the number of times of reproduction has surpassed the limit number of times. If the result of check is YES, the program moves to step S303 where the selected music number is deleted from the hard disc 15, at the same time as the information on the selected music number is deleted from the music number database. At step S304, the CPU 12 causes the CPU 32 to compute the new hash value of the music number database to store the computed hash value in the non-volatile memory 34. In this case, the playback output is inhibited.

[0088]

If, at step S302, the number of times of playback is less than the limit value, the program moves to step S305, where the CPU 12 verifies whether or not the date and time of playback end is past the current date and time. If the date and time of playback end is past the current date and time, the selected music number is deleted at step S303 from the hard disc, whilst the corresponding information is deleted from the music number database. At step S304, the hash values of the new music number database are computed at step S304 and stored. In this case, the playback output is inhibited.

[0089]

If it has been verified at step S305 that the date and time of playback end is not past the current date and time, the program moves to step S306, where the CPU 32 verifies whether or not the playback time charging condition has been added to the selected music number. If the playback time charging condition has been added to the

selected music number, the CPU 12 causes display of the message to the effect that the playback time charging condition has been added and the fee on the display 3. If it has been verified that the playback time charging condition has not been added to the selected music number, the processing at step S307 is skipped.

[0090]

Referring to the flowchart of Figs.26 and 27, the processing of outputting music data from the hard disc 15 via the memory stick walkman 6 is explained. At steps S321 to S325, the music number database is checked as to possible alteration of the music number database and the playback condition of the selected music number, whilst the selected music number is specified. This processing is not explained specifically since it is similar to that of steps S271 to S275 of Fig.23.

[0091]

At step S326, reciprocal authentication processing is executed between the memory stick walkman 6 and the personal computer 1 so that a session key is co-owned by the memory stick walkman 6 and the personal computer 1. At step S327, the CPU 12 of the personal computer 1 commands the memory stick walkman 6 to reproduce the encrypted audio data to be now sent thereto. At step S328, the CPU 12 reads out the filename of the selected music number specified at step S324 to read out the music data of the filename from the hard disc 15. The CPU 12 at step S329 executes the processing of converting the encoding system, encryption system or the format of the music data into those of the memory stick walkman 6. At step S330, the

CPU 12 encrypts the music data, converted at step S329, with the session key, to transfer the encrypted data to the memory stick walkman 6.

[0092]

At step S331, the authentication device 22 of the memory stick walkman 6 is responsive to the command transferred from the personal computer 1 to decode the transferred data with the session key to reproduce and output the decoded data. At step S332, the CPU 12 increments the number of times of playback count of the music number database by one. At step S333, the CPU 12 verifies whether or not the playback time charging condition has been added to the selected music number. If the result of check at step S333 is YES, the fee is written at step S334 in the charging log. At step S335, the CPU 12 causes the CPU 32 to newly compute and store the hash values of the entire music number database. If the result of check at step S333 is NO, that is if the playback time charging condition has not been added to the selected music number, the processing at steps S334 and S335 is skipped.

[0093]

According to the present invention, various elaborate techniques are used to prevent illicit duplication of the musical data. For example, the program for actuating the CPU 12 is a so-called tamper-resisting software in which the sequence of program execution is changed each time the program is executed.

[0094]

Moreover, part of the function of the CPU 12 is performed by the adapter 7 as

hardware, so that the various processing is executed by cooperation of the CPU and the adapter, thereby improving operational safety.

[0095]

For example, the hash values of the music number database are not saved in the music number database itself, but are saved in the non-volatile memory 34 of the adapter 7. That is, past hash values to be referred to for comparison with the hash values saved last time, such as at steps S32 or S33 of Fig.3, are saved in the non-volatile memory 34. Thus, it is possible to prevent copying or movement in which, before the musical data saved in the hard disc 15 are copied on or moved to another recording medium, the recording contents of the hard disc 15 are stored as backup data, and the latter data, kept as backup data in the hard disc 15, is re-stored after copying or moving the music data saved in the hard disc 15 to another recording medium .

[0096]

If, for example, the music numbers A and B are saved in the hard disc 15, hash values corresponding to the information of the music numbers A and B are saved in the non-volatile memory 34. It is now assumed that, in this state, recording data on the hard disc 15 are saved as backup in another recording medium. If, of the music numbers A and B saved in the hard disc 15, the music number A is moved to another recording medium 52, only the music number B is recorded on the hard disc 15. Thus, the hash value in the non-volatile memory 34 is also changed to that corresponding to

the music number B.

[0097]

Thus, if the contents of the hard disc 15 saved as backup on the recording medium 51 are re-stored on the hard disc 15 so that the music numbers A and B are again saved on the hard disc 15, the hash value computed from the information of the music number B is stored in the non-volatile memory 34, whilst the hash values computed from the music numbers A and B are not stored therein. Thus, at this time point, the hash value based on the music numbers A and B stored on the hard disc 15 and which is based on the music numbers A and B ceases to be coincident with the past hash value stored in the non-volatile memory 34, thus indicating that the music number database has been altered. The result is that limitations are subsequently imposed on the use of the music numbers A and B saved on the hard disc 15.

[0098]

Also, the RTC 35 is enclosed in the adapter 7. The time information in this RTC 35 is corrected based on the time information transferred from another device, such as EMD server 5, for which the correct result of authentication has been obtained. The current date and time used is not that supervised by the personal computer 1. Rather, the current date and time used is that outputted by the RTC 35. Thus, it is not possible for the user to correct the current time of the personal computer 1 to past time with inappropriate intention so as to evade the verification of the playback end time as the playback condition.

[0099]

Also, the adapter 7 is configured for decoding the encrypted and transferred program in accordance with the program stored in the ROM 36 from the outset to execute the program to improve operational safety. This will be explained by referring to the flowchart of Fig.29.

[0100]

That is, if it is desired for the personal computer 1 to execute pre-set processing on the adapter 7, the personal computer 1 at step S351 encrypts the program to be executed by the adapter 7, using the cryptographic key pre-stored in the memory 13, to transfer the encrypted program to the adapter 7. There is pre-stored in the ROM 36 of the adapter 7 the program for decoding and executing the encrypted program transferred from the personal computer 1. The CPU 32 at step S352 decodes the encrypted program, transferred from the personal computer 1, in accordance with the program stored in the ROM 36. The CPU 32 at step S313 expands the decoded program in the RAM 33. The CPU 32 at step S354 executes the expanded program.

[0101]

When the CPU 12 of the personal computer 1 causes the adapter 7 to compute the hash value of the music number database of the hard disc 15, the CPU 12 encrypts the data of the music number database with the cryptographic key to transfer the encrypted data to the CPU 32 of the adapter 7. The CPU 32 applies the hash function to the transferred data of the music number database to compute the hash value. The

computed hash value is stored in the non-volatile memory 34. Alternatively, the hash value is compared to the pre-stored past hash value to transfer the results of comparison to the CPU 12 of the personal computer 1.

[0102]

Fig.30 shows a more specified structure of the inside of the adapter 7 constructed as an semiconductor IC. The adapter 7 includes, in addition to the interface 31, CPU 32, RAM 33, non-volatile memory 34, RTC 35 and the ROM 36, shown in Fig.1, a RAM controller 61 for controlling the writing/readout to or from the RAM 33, and a logical circuit 62. The logical circuit 62 is used for performing processing of decoding the encrypted music data and outputting the decoded data directly from the adapter 7.

[0103]

The components from the interface 31 to the ROM 36, RAM controller 61 and the logical circuit 62 are integrally assembled into an semiconductor IC so as not to be dismounted from outside.

[0104]

A quartz oscillator 71 is used for generating reference clocks for the adapter 7 to execute various processing operations. An oscillating circuit 72 is used for actuating the RTC 35. A battery 73 furnished back-up power to the oscillating circuit 72, non-volatile memory 34 and to the RTC 35. To the remaining circuitry of the adapter 7 is supplied the power from a power source furnishing circuit 81 of the personal computer

1.

[0105]

The non-volatile memory 34 may be constructed by a write/erase ROM. However, if the non-volatile memory 34 is constructed by a RAM backed by the back-up power from the battery 73, it can be constructed by forming a protective aluminum layer 91 on the non-volatile memory 34 and by forming a power source pattern 92 thereon so as to be flush with the protective aluminum layer 91. The power source pattern 92 furnishes the power from the battery to the non-volatile memory 34. In this structure, if it is attempted to delete the protective aluminum layer 91 to alter the non-volatile memory 34, the power source pattern 34 lying flush with the protective aluminum layer 91 is also deleted to interrupt the power supply to the non-volatile memory 34 to erase the data stored in the inner region. Thus, with the present structure, tamper-resistant properties can be improved more effectively.

[0106]

Moreover, data write/readout interconnections 101-1 to 101-3 for the non-volatile memory 34 are arranged in vertically aligned and reciprocally separated positions, as shown in Fig.32. Thus, in order to read out data from the lower layer interconnection 101-3, the upper interconnections 101-1 and 101-2 need to be removed, with the result that data cannot be read out simultaneously from the interconnections 101-1 to 101-3. In addition, if the interconnections 101-1 to 101-3 are formed redundantly, and probed directly, it becomes possible to render it difficult

to analyze the contents by the added capacity.

[0107]

In the above-described arrangement, the memory stick walkman 6 is used as a recording medium. However, the present invention may also be applied for transferring or copying data to recording mediums other than the memory stick walkman.

[0108]

The data may also be picture or other data, in addition to music data.

[0109]

The present invention gives rise to the following meritorious effects:

[0110]

(1) Since data can be recorded in an encrypted form on the hard disc 15and moreover the cryptographic key is encrypted with the storage key and subsequently recorded on the hard disc 15, the music data recorded on the hard disc 15, if copied, cannot be decoded, so that it is possible to prevent distribution of a large quantity of duplications.

[0111]

(2) If a music number is copied once, the music number and the date and time of recording is registered in the database, in order that the music number, once copied, cannot be copied for a pre-set time, which is 48 hours in the illustrated embodiment. It is thus possible to limit the number of times of copying and hence to have copies distributed in large quantities.

[0112]

Moreover, the data hash values are calculated and saved every time the database is updated, to render it possible to prevent the database from being altered.

[0113]

(3) If music data is delivered to an external equipment, the music data on the hard disc 15 is erased, so that original digital music data is not left on the hard disc 15 to render it possible to prevent the copies from being distributed in large quantities.

[0114]

(4) A music number database is provided on the hard disc 15, and the hash values of the entire database are checked each time. Thus, if the contents of the hard disc 15 are saved as backup before moving the contents and it is then attempted to re-store the backup data directly after the movement, the source data can be erased reliably.

[0115]

(5) Since reciprocal authentication processing is carried out before delivering data from the personal computer 1 to the external equipment, it is possible to prevent data from being delivered to an unauthorized equipment.

[0116]

(6) Before delivering data from an external equipment to the personal computer 1, reciprocal authentication processing is carried out to check whether or not the software of the personal computer 1 is authorized. Thus, it is possible to prevent music data from being delivered to an unauthorized software.

[0117]

(7) Since ISRC is used for verifying identity of the music number and, if the ISRC is not acquired, TOC is used, the identity of the music number can be verified even if the ISRC cannot be acquired.

[0118]

(8) Since a pre-set portion of the software function in the personal computer 1 is performed by the adapter 7 annexed to the personal computer 1, simple analysis of the software of the personal computer 1 does not clarify the nature of processing in its entirety to render it difficult to alter the software to produce the function in meeting with the inappropriate intention.

[0119]

It is noted that the system in the context of the present specification means an ensemble of plural devices.

[0120]

The furnishing mediums for furnishing a computer program performing the above processing may be a communication medium, such as network or satellite, in addition to the magnetic disc, CD-ROM or a solid-state memory.

[0121]

[Effect of the Invention]

As described above, with the above information processing apparatus according to Claim 1, the information method according to Claim 4 and the information

furnishing medium according to Claim 5, the controlling means comprised of a software controls storage or read of data into or from the data storage means based on the result of the program execution by a program executing means incorporated in a hardware. Thus, it is possible to prevent falsification of a software used with data in order to inhibit fraudulent copying of the data.

[Brief Description of the Drawings]

Fig.1 is a block diagram showing an illustrative structure of a system embodying the present invention.

Fig.2 is a flowchart for illustrating the processing for copying from a compact disc to a hard disc 15.

Fig.3 is a flowchart for illustrating the terminal database check processing of step S12 of Fig.2.

Fig.4 shows an illustrative terminal database.

Fig.5 illustrates a watermark.

Fig.6 shows an illustrative music number database.

Fig.7 is a flowchart for illustrating the operation of moving data from the hard disc 15 of the system of Fig.1 to the memory stick walkman 6.

Fig.8, continuing to Fig.7, is a flowchart for illustrating the operation of moving data from the hard disc 15 of the system of Fig.1 to the memory stick walkman 6.

Fig.9, continuing to Fig.8, is a flowchart for illustrating the operation of moving data from the hard disc 15 of the system of Fig.1 to the memory stick walkman 6.

Fig.10 is a flowchart for illustrating the check processing for e.g., playback conditions for the music number selected at step S55 of Fig.7

Fig.11 illustrates the playback conditions supervised by the memory stick walkman.

Fig.12 is a flowchart for illustrating details of the format conversion processing of step S58 of Fig.7.

Fig.13 is a flowchart for illustrating the operation of copying data from the hard disc 15 of Fig.1 to the memory stick walkman 6.

Fig.14, continuing to Fig.13, is a is a flowchart for illustrating the operation of copying data from the hard disc 15 of Fig.1 to the memory stick walkman 6.

Fig.15, continuing to Fig.14, is a is a flowchart for illustrating the operation of copying data from the hard disc 15 of Fig.1 to the memory stick walkman 6.

Fig.16 is a flowchart for illustrating the operation of moving data from the memory stick walkman 6 of Fig.1 to the hard disc 15.

Fig.17 is a flowchart for illustrating the operation of copying data from the memory stick walkman 6 of Fig.1 to the hard disc 15.

Fig.18 is a flowchart for illustrating the operation of copying data from an EMD server 5 of the system of Fig.1 to the hard disc 15.

Fig.19 is a flowchart for illustrating details of the charging processing of step S204 of Fig.18.

Fig.20 illustrates a charging log.

Fig.21 is a flowchart for illustrating the operation of copying data from an IEC60958 terminal 16a of the system of Fig.1 to the hard disc 15.

Fig.22 is a flowchart for illustrating the operation of copying data from the IEC60958 terminal 16a of the system of Fig.1 to the hard disc 15.

Fig.23 is a flowchart for illustrating the operation of outputting data from the hard disc 15 of the system of Fig.1 to the IEC60958 terminal 16a.

Fig.24, continuing to Fig.23, is a flowchart for illustrating the operation of outputting data from the hard disc 15 of the system of Fig.1 to the IEC60958 terminal 16a.

Fig.25 is a flowchart for illustrating the processing for checking the playback conditions of step S275 of Fig.23.

Fig.26 is a flowchart for illustrating the processing of outputting from the hard disc 15 of the system of Fig.1 via the memory stick walkman 6.

Fig.27, continuing to Fig.26, is a flowchart for illustrating the processing of outputting from the hard disc 15 of the system of Fig.1 via the memory stick walkman 6.

Fig.28 illustrates the function of a non-volatile memory 34 of Fig.1.

Fig.29 is a flowchart for illustrating the operation of an adapter 7 of the system of Fig.1.

Fig.30 illustrates the inner structure of the adapter 7 of the system of Fig.1.

Fig.31 shows an illustrative inner structure of the non-volatile memory 34 of

Fig.31.

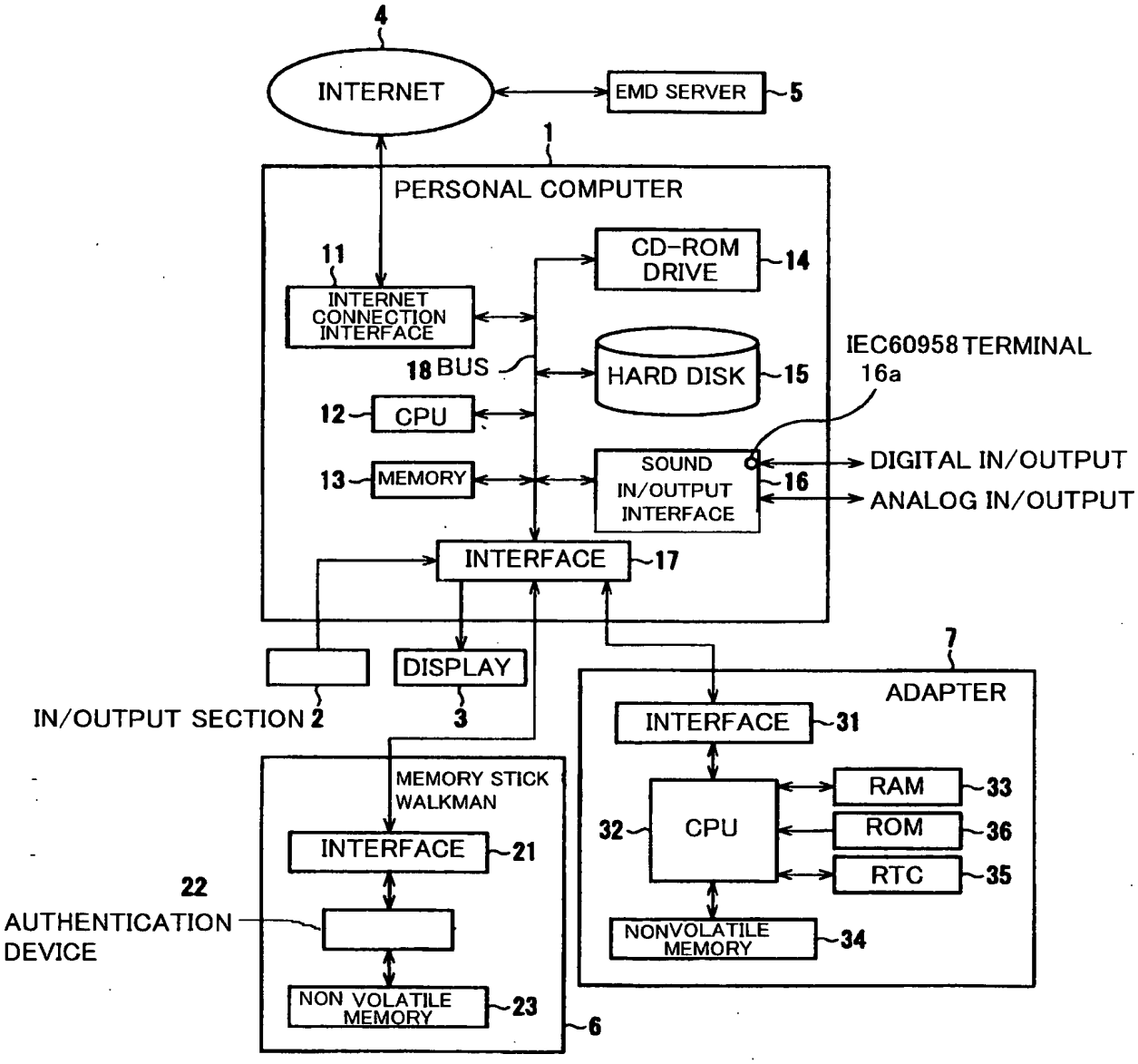
Fig.32 also shows an illustrative inner structure of the non-volatile memory 34 of Fig.31.

[Explanation of Referenced Numerals]

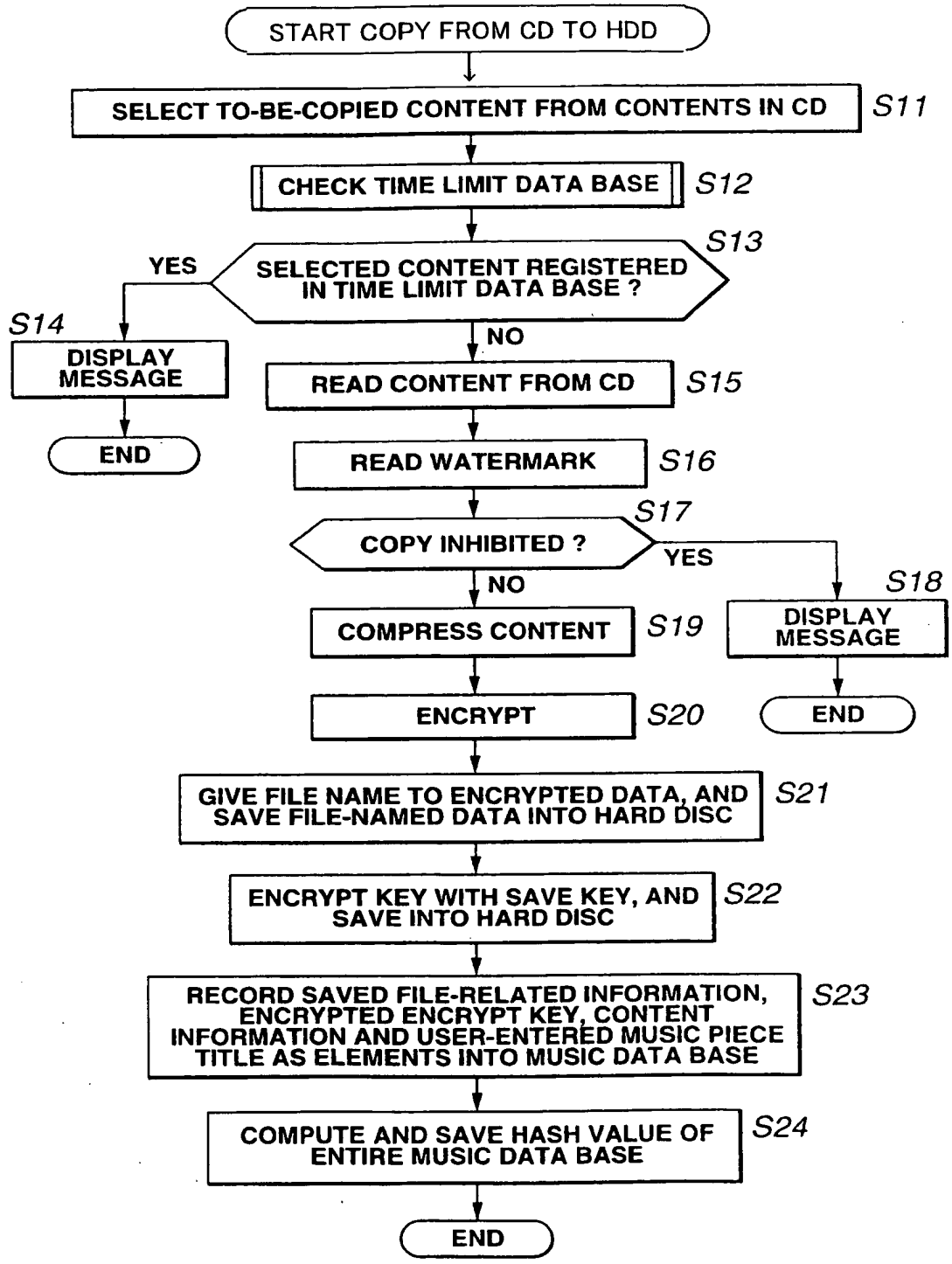
1 personal computer; 2 input unit; 3 display; 4 internet; 5 EMD server; 6 memory stick walkman; 7 adapter; 12 CPU; 13 memory; 14 CD-ROM drive; 15 hard disc; 16 audio input/output interface; 16a IEC 60958 terminal; 22 authentication device; 23 non-volatile memory; 32 CPU

[DOCUMENT NAME] DRAWING

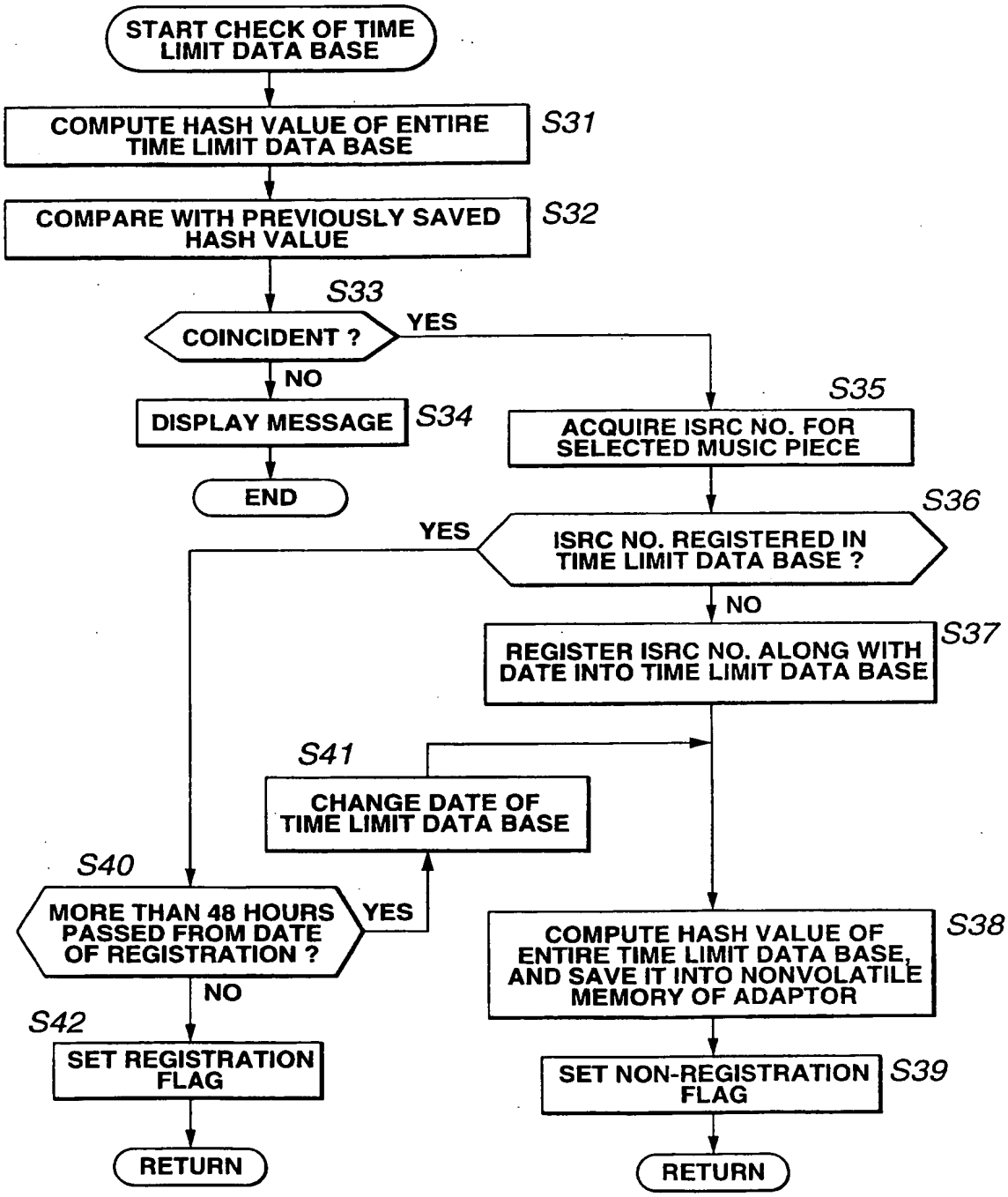
[FIG. 1]



[FIG. 2]


START COPY FROM CD TO HDD
SELECT TO-BE-COPIED CONTENT FROM CONTENTS IN CD
S11
CHECK TIME LIMIT DATA BASE
S12
S13
SELECTED CONTENT REGISTERED IN TIME LIMIT DATA BASE ?
YES
NO
S14
DISPLAY MESSAGE
END
READ CONTENT FROM CD
S15
READ WATERMARK
S16
S17
COPY INHIBITED ?
YES
NO
S18
DISPLAY MESSAGE
END
COMPRESS CONTENT
S19
ENCRYPT
S20
GIVE FILE NAME TO ENCRYPTED DATA, AND SAVE FILE-NAMED DATA INTO HARD DISC
S21
ENCRYPT KEY WITH SAVE KEY, AND SAVE INTO HARD DISC
S22
RECORD SAVED FILE-RELATED INFORMATION, ENCRYPTED ENCRYPT KEY, CONTENT INFORMATION AND USER-ENTERED MUSIC PIECE TITLE AS ELEMENTS INTO MUSIC DATA BASE
S23
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S24
END

[FIG. 3]


START CHECK OF TIME LIMIT DATA BASE
COMPUTE HASH VALUE OF ENTIRE TIME LIMIT DATA BASE
S31
COMPARE WITH PREVIOUSLY SAVED HASH VALUE
S32
S33
COINCIDENT ?
YES
NO
DISPLAY MESSAGE
S34
END
S35
ACQUIRE ISRC NO. FOR SELECTED MUSIC PIECE
S36
YES
ISRC NO. REGISTERED IN TIME LIMIT DATA BASE ?
NO
REGISTER ISRC NO. ALONG WITH DATE INTO TIME LIMIT DATA BASE
S37
S41
CHANGE DATE OF TIME LIMIT DATA BASE
S40
MORE THAN 48 HOURS PASSED FROM DATE OF REGISTRATION ?
YES
NO
S42
SET REGISTRATION FLAG
RETURN
COMPUTE HASH VALUE OF ENTIRE TIME LIMIT DATA BASE, AND SAVE IT INTO NONVOLATILE MEMORY OF ADAPTOR
S38
SET NON-REGISTRATION FLAG
S39
RETURN

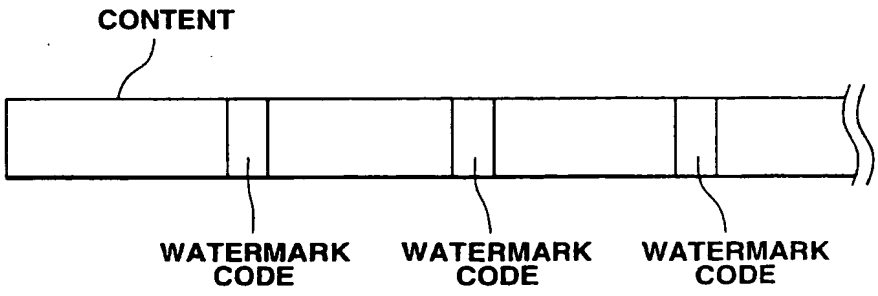
[FIG. 4]

TIME LIMIT DATA BASE

	ITEM 1	ITEM 2	ITEM 3	
ISRC	JP-Z90-98-12345	US-Z90-99-12346	JP-Z90-98-12347	
DATE OF COPY	1998.11.23.08:04	2004.03.06.16:09	2004.03.06.16.15	

HASH VALUE	0xf3352e125934

[FIG. 5]



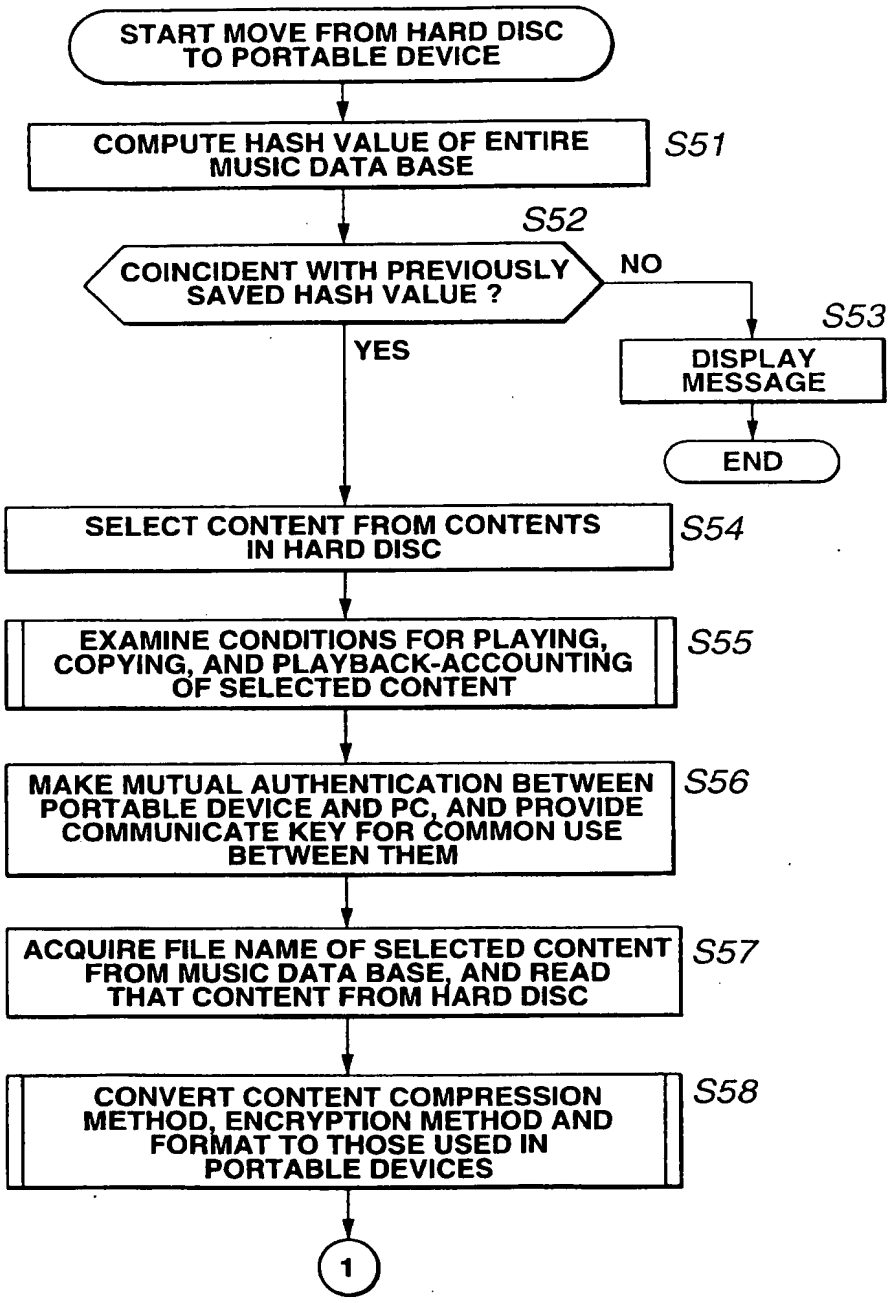
[FIG. 6]

MUSIC DATA BASE

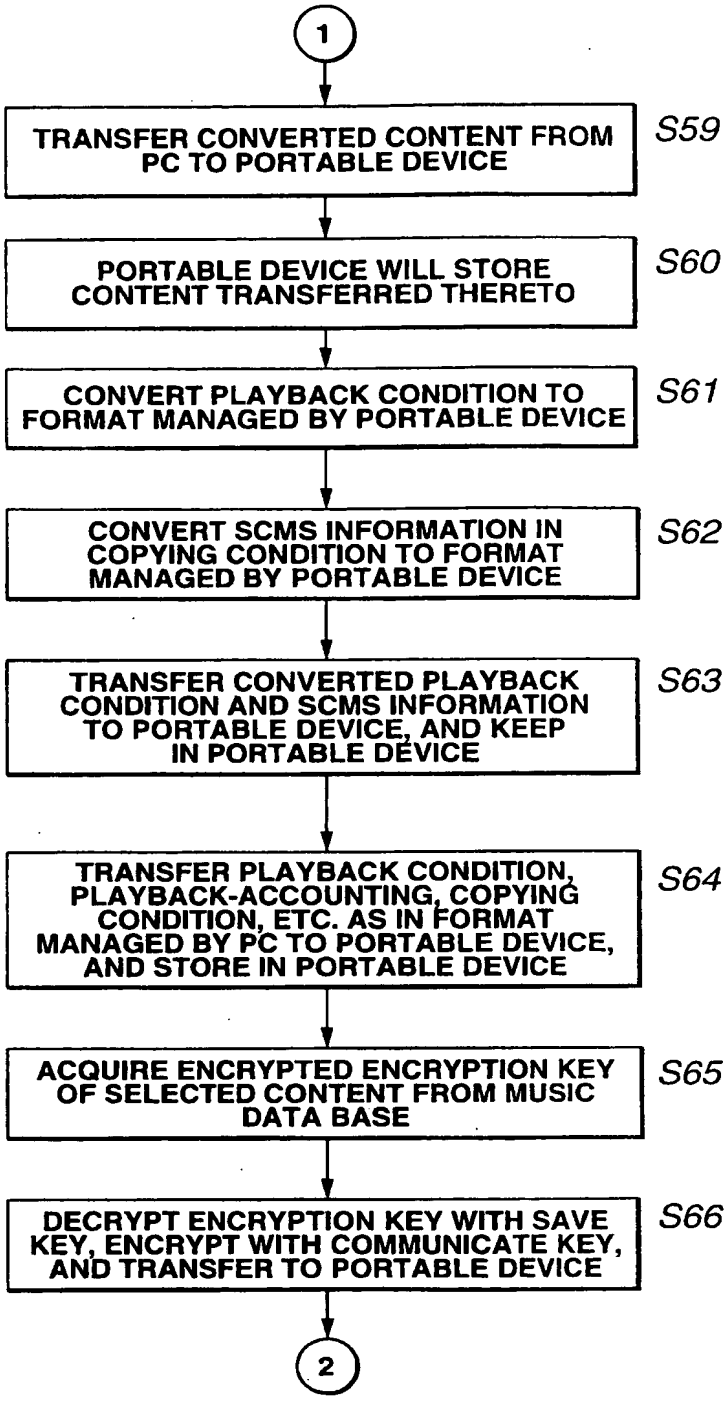
	ITEM 1	ITEM 2	ITEM 3	
FILE NAME	Xd000110. at2	px92341234. at2	aa0234287034. at2	
ENCRYPTED ENCRYPTION KEY	0xabababababab	0x9898989898989	0x123456789012	
MUSIC PIECE NAME	HARU-NO-OGAWA	UNMEI(DESTINY)	KOUJOU-NO-TSUKI	
PLAY TIME LENGTH	180	190	200	
PLAYBACK CONDITION : START DATE	-	2001.01.01.00:00	-	
PLAYBACK CONDITION : END DATE	1999.07.31.23:59	-	-	
PLAYBACK CONDITION : PLAYBACK LIMIT	-	20	-	
PLAYBACK COUNTER	-	12	-	
PLAYBACK ACCOUNTING CONDITION	-	-	¥5	
COPYING CONDITION : COPIES	2	0	0	
COPY COUNTER	1	0	0	
COPYING CONDITION : SCMS	0b01	0b10	0b00	

HASH VALUE	0xf9951e566321

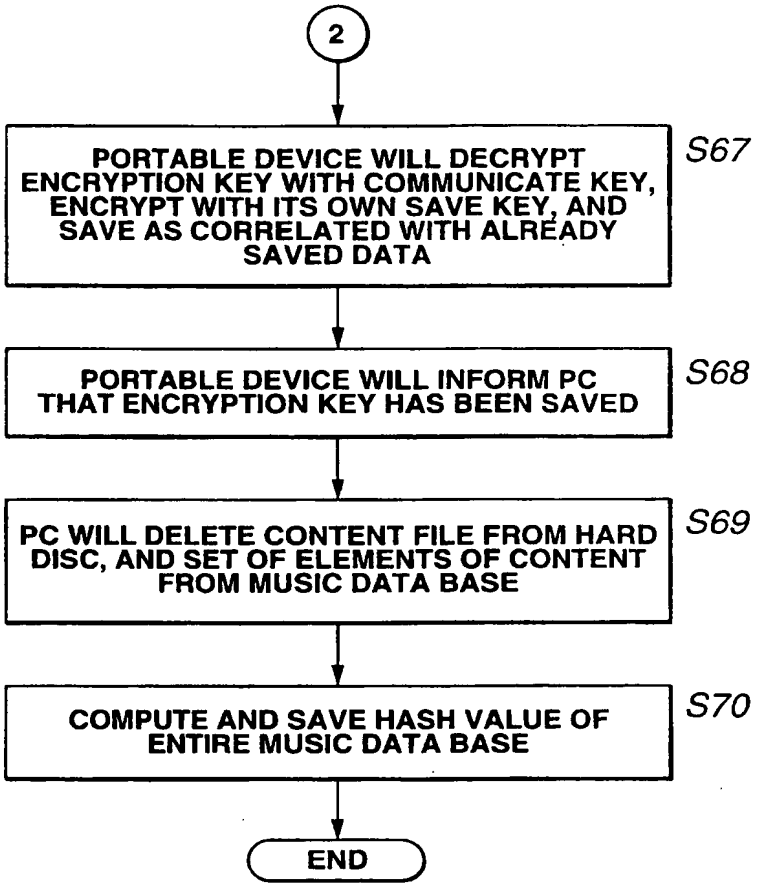
[FIG. 7]


START MOVE FROM HARD DISC TO PORTABLE DEVICE
COMPUTE HASH VALUE OF ENTIRE MUSIC DATA BASE
S51
S52
COINCIDENT WITH PREVIOUSLY SAVED HASH VALUE ?
NO
YES
S53
DISPLAY MESSAGE
END
SELECT CONTENT FROM CONTENTS IN HARD DISC
S54
EXAMINE CONDITIONS FOR PLAYING, COPYING, AND PLAYBACK-ACCOUNTING OF SELECTED CONTENT
S55
MAKE MUTUAL AUTHENTICATION BETWEEN PORTABLE DEVICE AND PC, AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S56
ACQUIRE FILE NAME OF SELECTED CONTENT FROM MUSIC DATA BASE, AND READ THAT CONTENT FROM HARD DISC
S57
CONVERT CONTENT COMPRESSION METHOD, ENCRYPTION METHOD AND FORMAT TO THOSE USED IN PORTABLE DEVICES
S58
1

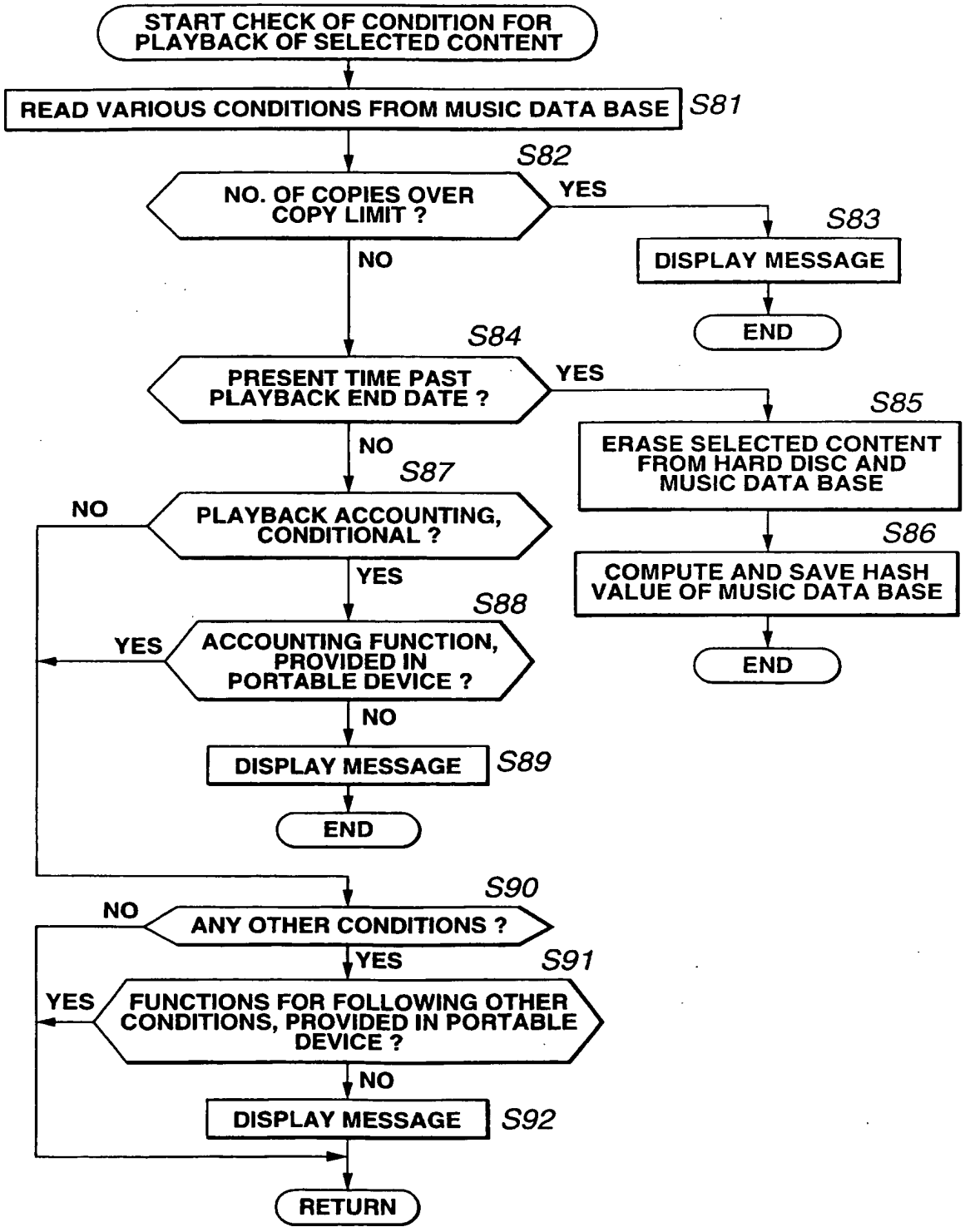
[FIG. 8]


1
TRANSFER CONVERTED CONTENT FROM PC TO PORTABLE DEVICE
S59
PORTABLE DEVICE WILL STORE CONTENT TRANSFERRED THERETO
S60
CONVERT PLAYBACK CONDITION TO FORMAT MANAGED BY PORTABLE DEVICE
S61
CONVERT SCMS INFORMATION IN COPYING CONDITION TO FORMAT MANAGED BY PORTABLE DEVICE
S62
TRANSFER CONVERTED PLAYBACK CONDITION AND SCMS INFORMATION TO PORTABLE DEVICE, AND KEEP IN PORTABLE DEVICE
S63
TRANSFER PLAYBACK CONDITION, PLAYBACK-ACCOUNTING, COPYING CONDITION, ETC. AS IN FORMAT MANAGED BY PC TO PORTABLE DEVICE, AND STORE IN PORTABLE DEVICE
S64
ACQUIRE ENCRYPTED ENCRYPTION KEY OF SELECTED CONTENT FROM MUSIC DATA BASE
S65
DECRYPT ENCRYPTION KEY WITH SAVE KEY, ENCRYPT WITH COMMUNICATE KEY, AND TRANSFER TO PORTABLE DEVICE
S66
2

[FIG. 9]


2
PORTABLE DEVICE WILL DECRYPT ENCRYPTION KEY WITH COMMUNICATE KEY, ENCRYPT WITH ITS OWN SAVE KEY, AND SAVE AS CORRELATED WITH ALREADY SAVED DATA
S67
PORTABLE DEVICE WILL INFORM PC THAT ENCRYPTION KEY HAS BEEN SAVED
S68
PC WILL DELETE CONTENT FILE FROM HARD DISC, AND SET OF ELEMENTS OF CONTENT FROM MUSIC DATA BASE
S69
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S70
END

[FIG. 10]

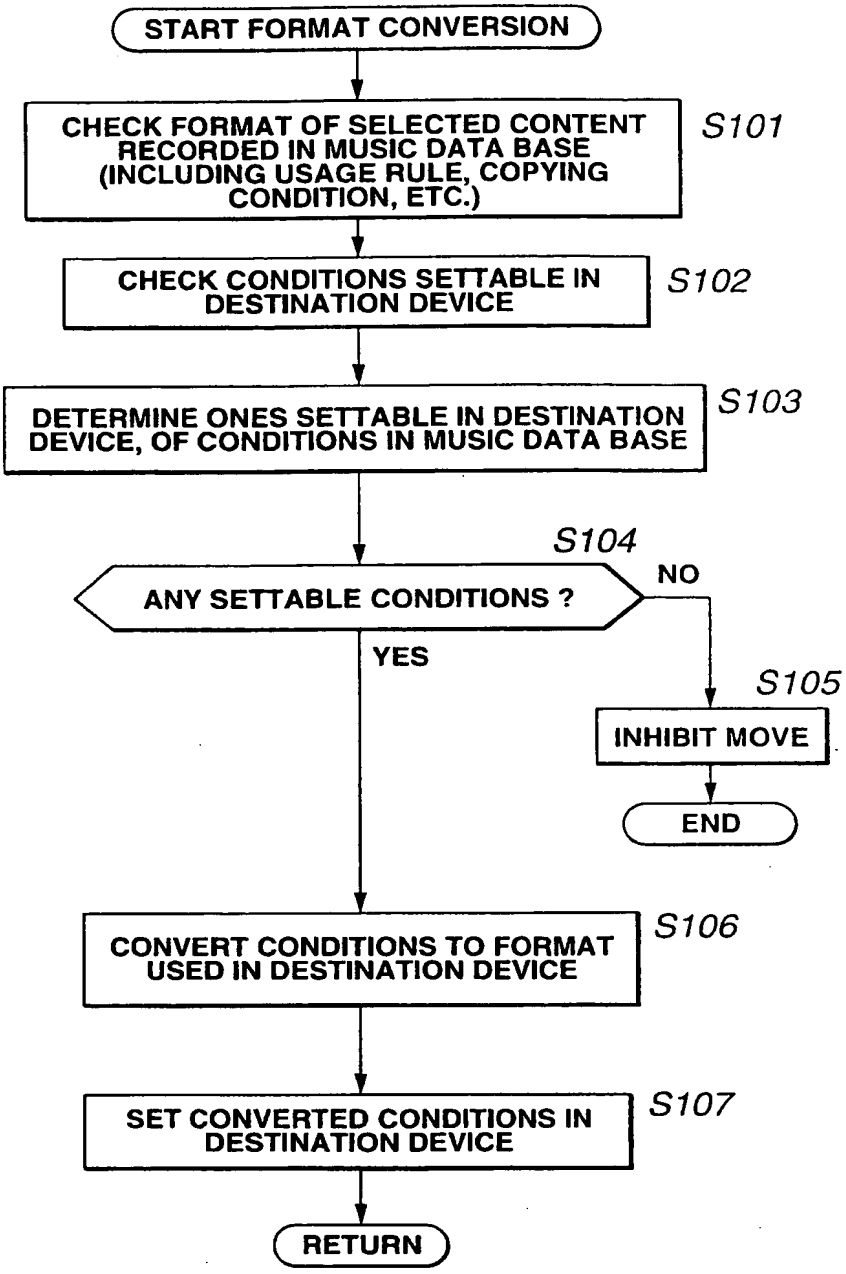

START CHECK OF CONDITION FOR PLAYBACK OF SELECTED CONTENT
READ VARIOUS CONDITIONS FROM MUSIC DATA BASE
S81
S82
NO. OF COPIES OVER COPY LIMIT ?
YES
S83
DISPLAY MESSAGE
END
NO
S84
PRESENT TIME PAST PLAYBACK END DATE ?
YES
S85
ERASE SELECTED CONTENT FROM HARD DISC AND MUSIC DATA BASE
S86
COMPUTE AND SAVE HASH VALUE OF MUSIC DATA BASE
END
NO
S87
PLAYBACK ACCOUNTING, CONDITIONAL ?
YES
S88
YES
ACCOUNTING FUNCTION, PROVIDED IN PORTABLE DEVICE ?
NO
DISPLAY MESSAGE
S89
END
S90
NO
ANY OTHER CONDITIONS ?
YES
S91
YES
FUNCTIONS FOR FOLLOWING OTHER CONDITIONS, PROVIDED IN PORTABLE DEVICE ?
NO
DISPLAY MESSAGE
S92
RETURN

[FIG. 11]

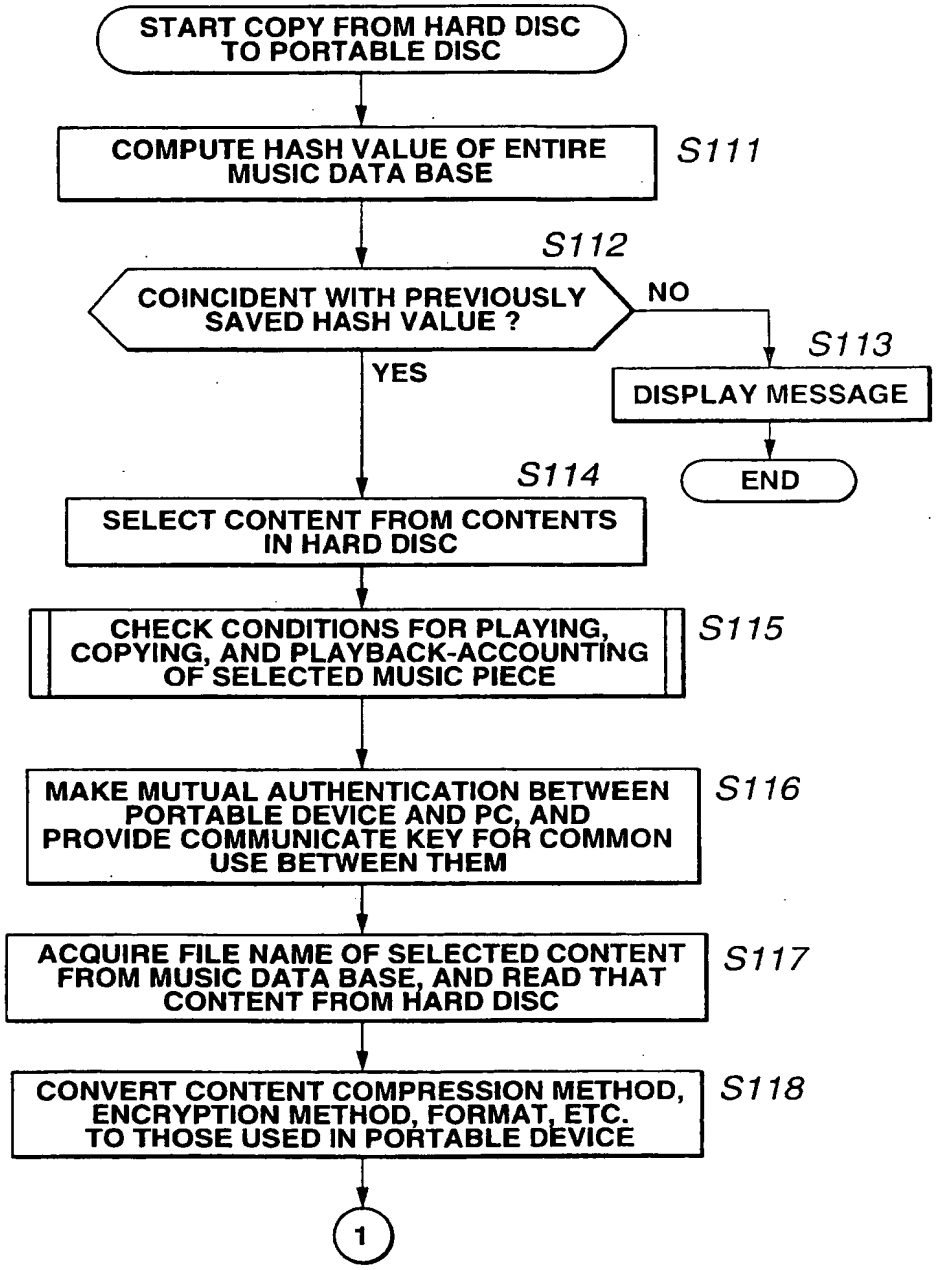
PLAYBACK CONDITIONS MANAGED BY PORTABLE DEVICE

	ITEM 1	ITEM 2	ITEM 3
CONTENT ID	00001	00002	00003
PLAY START DATE	1999.07.31.23:59	1999.07.31.23:59	1999.07.31.23:59
PLAY END DATE	2001.01.01.00:00	2001.01.01.00:00	2001.01.01.00:00
PLAYBACK LIMIT	-	15	-

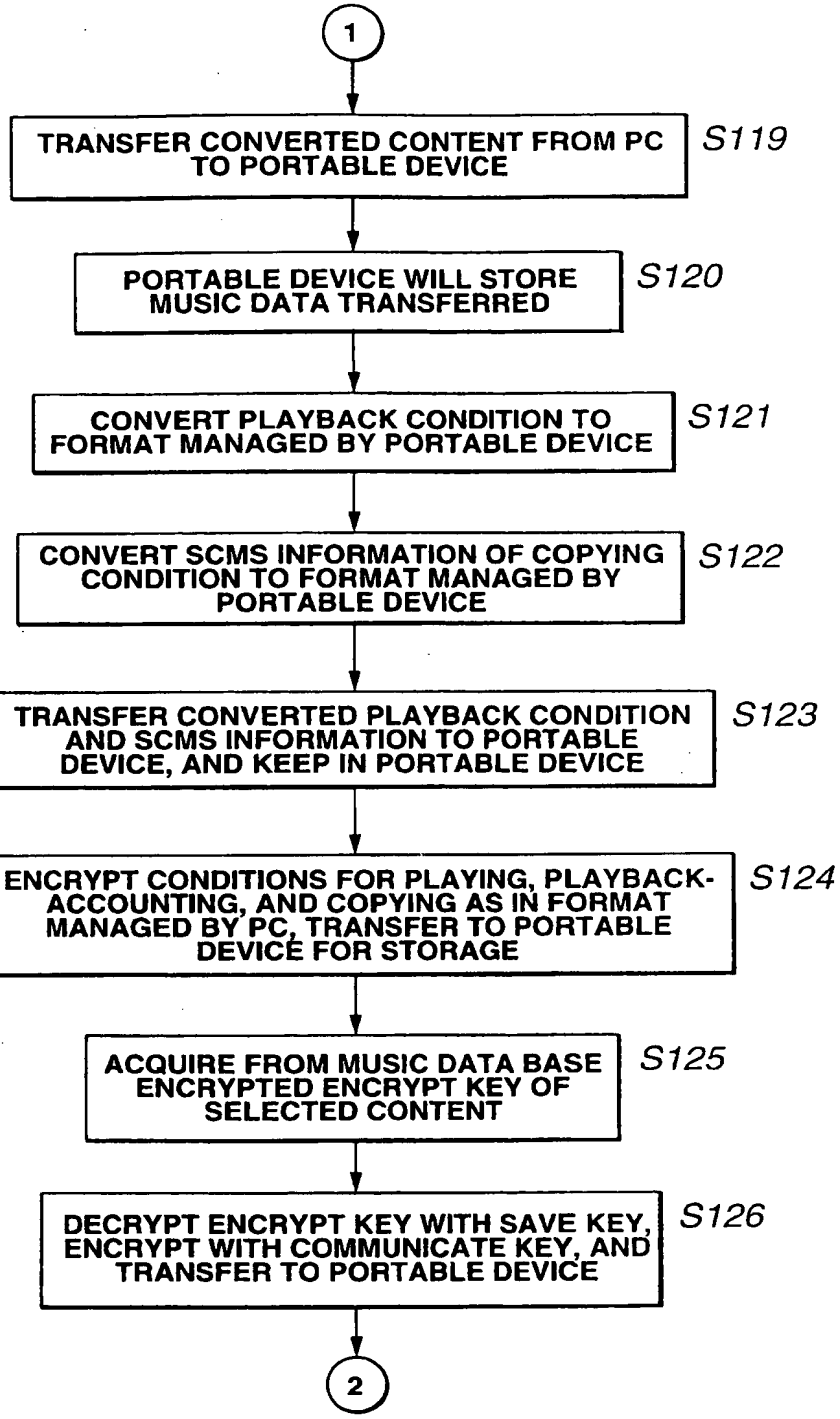
[FIG. 12]


START FORMAT CONVERSION
CHECK FORMAT OF SELECTED CONTENT RECORDED IN MUSIC DATA BASE (INCLUDING USAGE RULE, COPYING CONDITION, ETC.)
S101
CHECK CONDITIONS SETTABLE IN DESTINATION DEVICE
S102
DETERMINE ONES SETTABLE IN DESTINATION DEVICE, OF CONDITIONS IN MUSIC DATA BASE
S103
S104
ANY SETTABLE CONDITIONS ?
NO
YES
S105
INHIBIT MOVE
END
CONVERT CONDITIONS TO FORMAT USED IN DESTINATION DEVICE
S106
SET CONVERTED CONDITIONS IN DESTINATION DEVICE
S107
RETURN

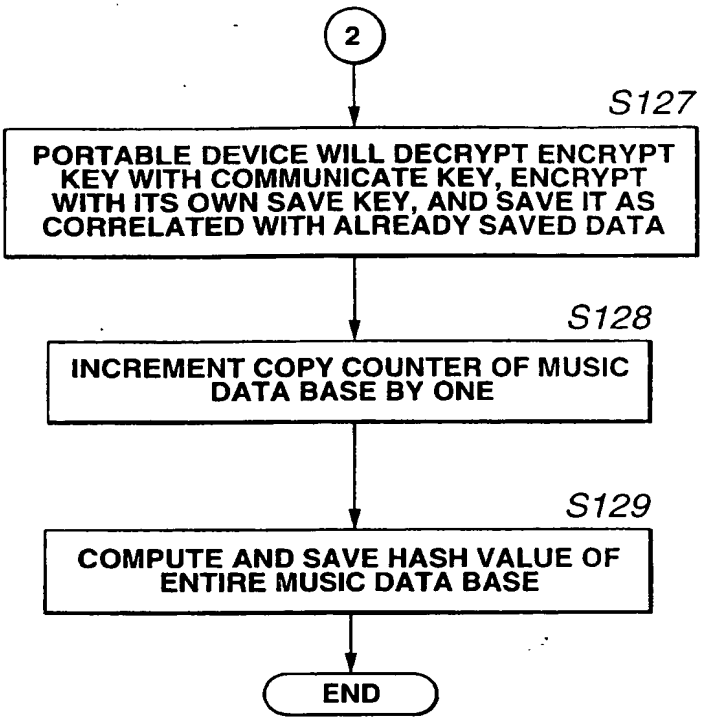
[FIG. 13]


START COPY FROM HARD DISC TO PORTABLE DISC
COMPUTE HASH VALUE OF ENTIRE MUSIC DATA BASE
S111
S112
COINCIDENT WITH PREVIOUSLY SAVED HASH VALUE ?
NO
S113
DISPLAY MESSAGE
END
YES
S114
SELECT CONTENT FROM CONTENTS IN HARD DISC
CHECK CONDITIONS FOR PLAYING, COPYING, AND PLAYBACK-ACCOUNTING OF SELECTED MUSIC PIECE
S115
MAKE MUTUAL AUTHENTICATION BETWEEN PORTABLE DEVICE AND PC, AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S116
ACQUIRE FILE NAME OF SELECTED CONTENT FROM MUSIC DATA BASE, AND READ THAT CONTENT FROM HARD DISC
S117
CONVERT CONTENT COMPRESSION METHOD, ENCRYPTION METHOD, FORMAT, ETC. TO THOSE USED IN PORTABLE DEVICE
S118
1

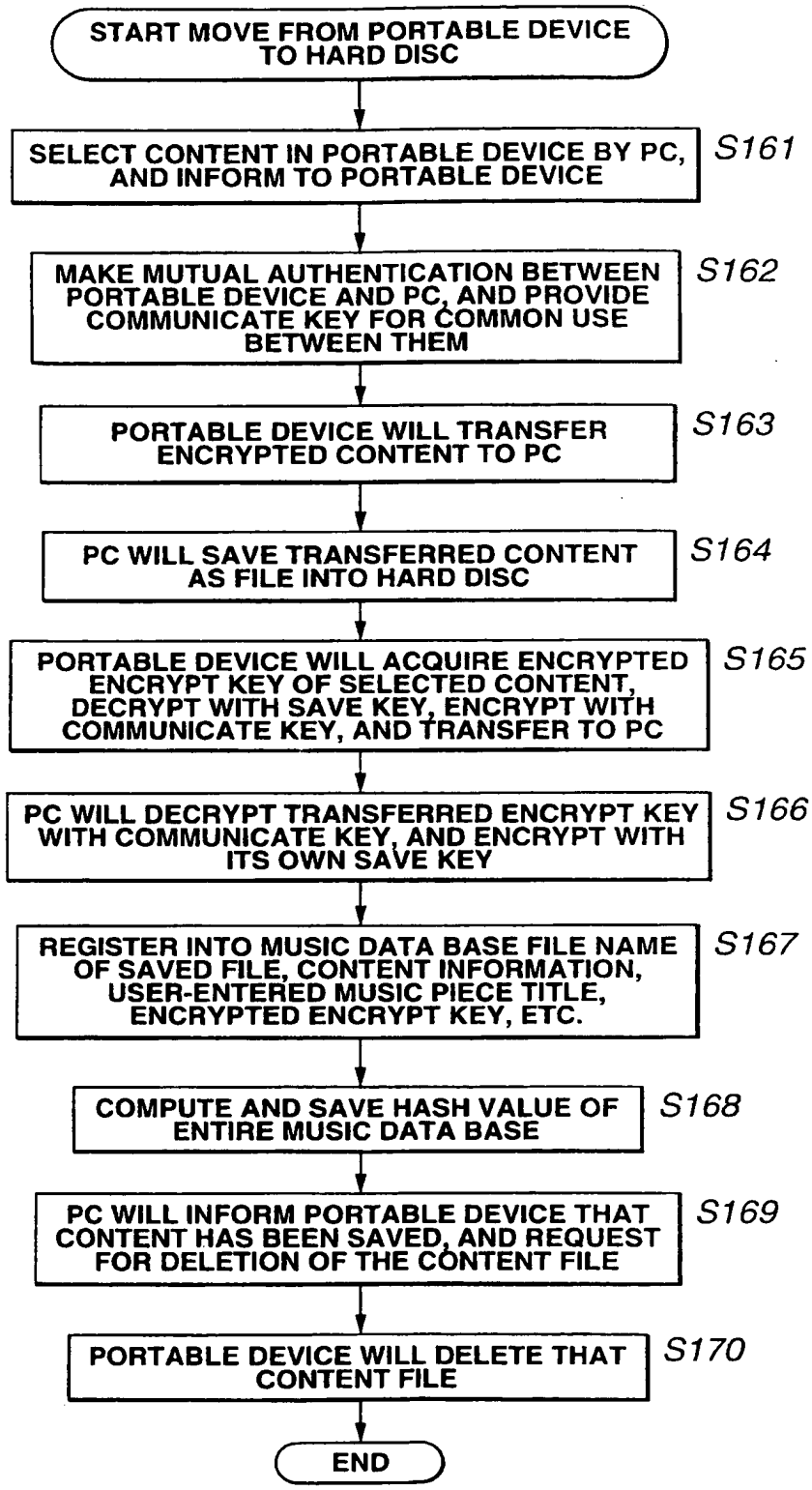
[FIG. 14]


1
TRANSFER CONVERTED CONTENT FROM PC TO PORTABLE DEVICE
S119
PORTABLE DEVICE WILL STORE MUSIC DATA TRANSFERRED
S120
CONVERT PLAYBACK CONDITION TO FORMAT MANAGED BY PORTABLE DEVICE
S121
CONVERT SCMS INFORMATION OF COPYING CONDITION TO FORMAT MANAGED BY PORTABLE DEVICE
S122
TRANSFER CONVERTED PLAYBACK CONDITION AND SCMS INFORMATION TO PORTABLE DEVICE, AND KEEP IN PORTABLE DEVICE
S123
ENCRYPT CONDITIONS FOR PLAYING, PLAYBACK-ACCOUNTING, AND COPYING AS IN FORMAT MANAGED BY PC, TRANSFER TO PORTABLE DEVICE FOR STORAGE
S124
ACQUIRE FROM MUSIC DATA BASE ENCRYPTED ENCRYPT KEY OF SELECTED CONTENT
S125
DECRYPT ENCRYPT KEY WITH SAVE KEY, ENCRYPT WITH COMMUNICATE KEY, AND TRANSFER TO PORTABLE DEVICE
S126
2

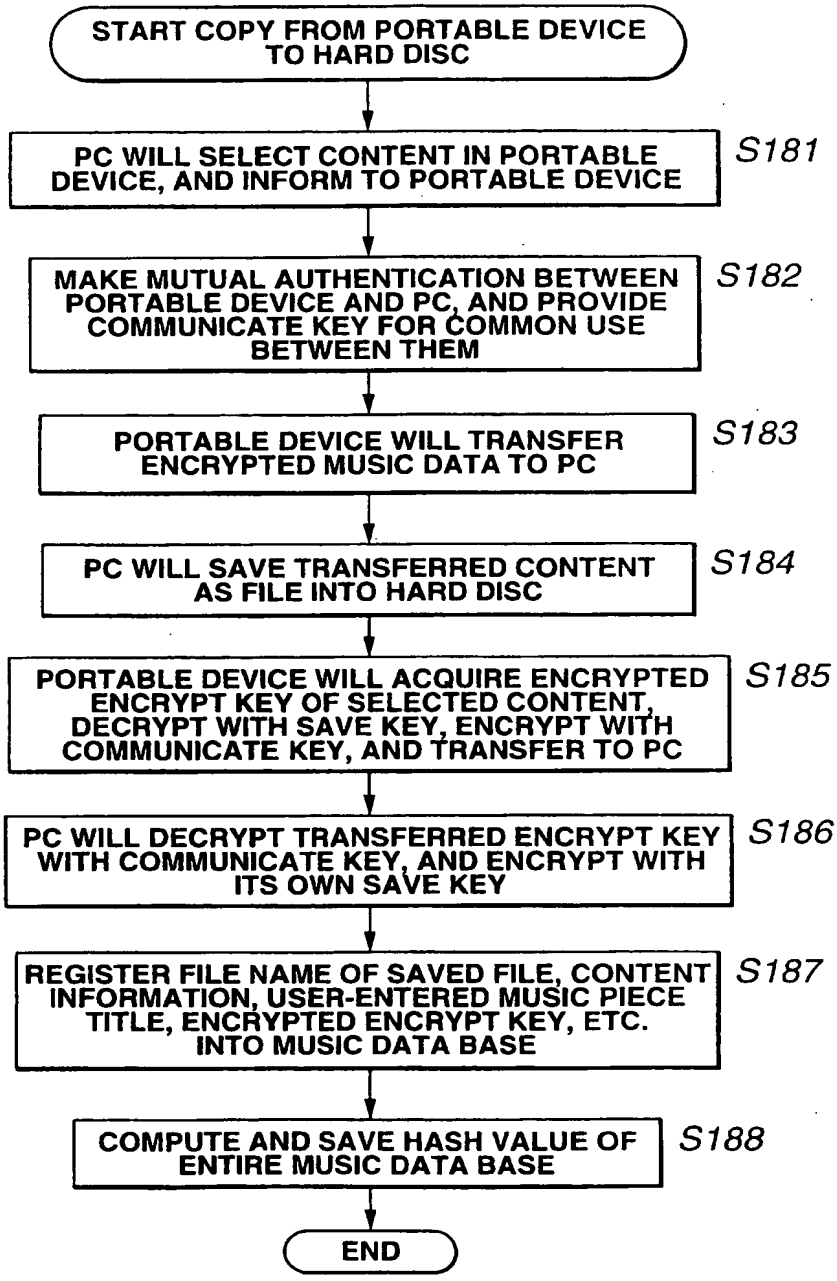
[FIG. 15]


2
S127
PORTABLE DEVICE WILL DECRYPT ENCRYPT KEY WITH COMMUNICATE KEY, ENCRYPT WITH ITS OWN SAVE KEY, AND SAVE IT AS CORRELATED WITH ALREADY SAVED DATA
S128
INCREMENT COPY COUNTER OF MUSIC DATA BASE BY ONE
S129
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
END

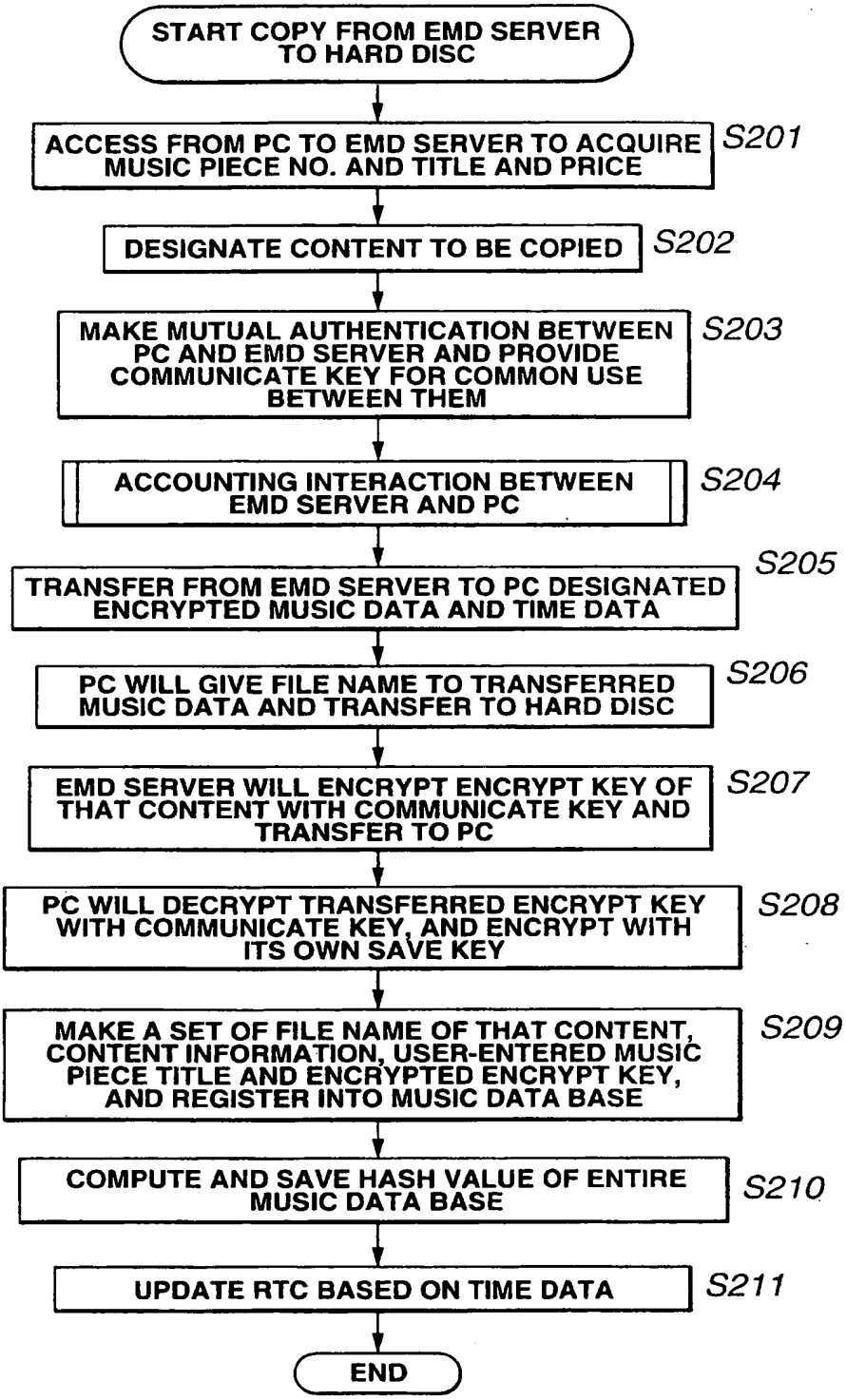
[FIG. 16]


START MOVE FROM PORTABLE DEVICE TO HARD DISC
SELECT CONTENT IN PORTABLE DEVICE BY PC, AND INFORM TO PORTABLE DEVICE
S161
MAKE MUTUAL AUTHENTICATION BETWEEN PORTABLE DEVICE AND PC, AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S162
PORTABLE DEVICE WILL TRANSFER ENCRYPTED CONTENT TO PC
S163
PC WILL SAVE TRANSFERRED CONTENT AS FILE INTO HARD DISC
S164
PORTABLE DEVICE WILL ACQUIRE ENCRYPTED ENCRYPT KEY OF SELECTED CONTENT, DECRYPT WITH SAVE KEY, ENCRYPT WITH COMMUNICATE KEY, AND TRANSFER TO PC
S165
PC WILL DECRYPT TRANSFERRED ENCRYPT KEY WITH COMMUNICATE KEY, AND ENCRYPT WITH ITS OWN SAVE KEY
S166
REGISTER INTO MUSIC DATA BASE FILE NAME OF SAVED FILE, CONTENT INFORMATION, USER-ENTERED MUSIC PIECE TITLE, ENCRYPTED ENCRYPT KEY, ETC.
S167
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S168
PC WILL INFORM PORTABLE DEVICE THAT CONTENT HAS BEEN SAVED, AND REQUEST FOR DELETION OF THE CONTENT FILE
S169
PORTABLE DEVICE WILL DELETE THAT CONTENT FILE
S170
END

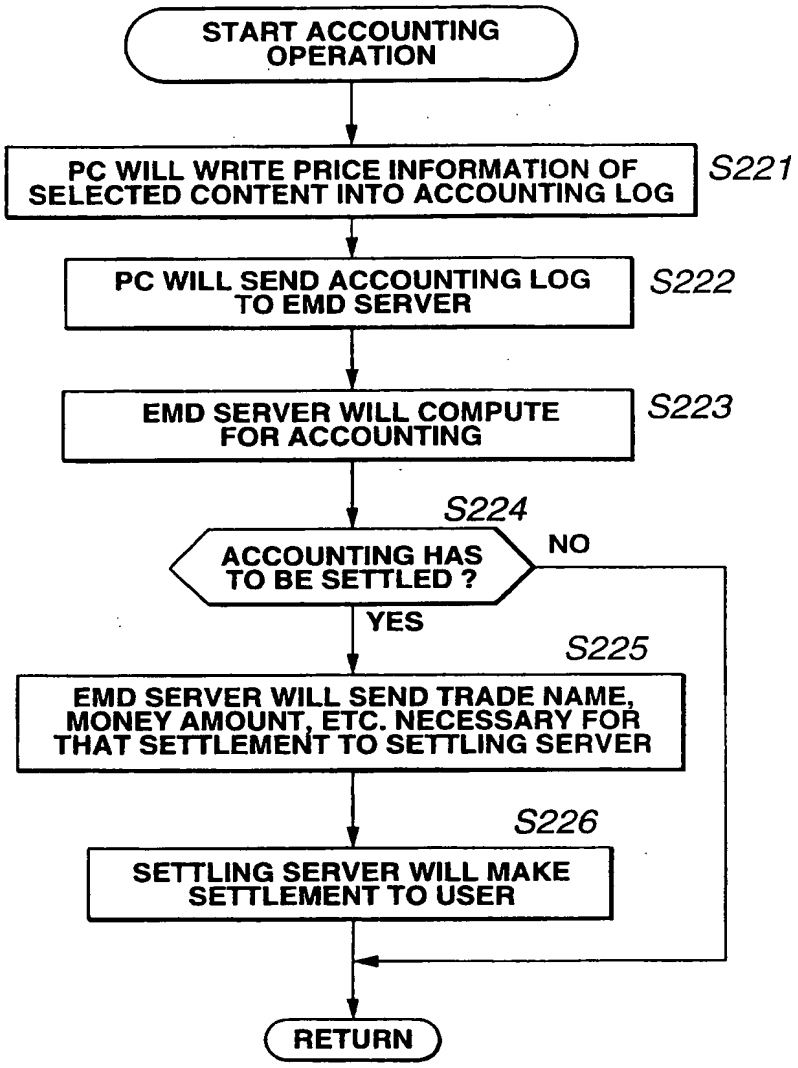
[FIG. 17]


START COPY FROM PORTABLE DEVICE TO HARD DISC
PC WILL SELECT CONTENT IN PORTABLE DEVICE, AND INFORM TO PORTABLE DEVICE
S181
MAKE MUTUAL AUTHENTICATION BETWEEN PORTABLE DEVICE AND PC, AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S182
PORTABLE DEVICE WILL TRANSFER ENCRYPTED MUSIC DATA TO PC
S183
PC WILL SAVE TRANSFERRED CONTENT AS FILE INTO HARD DISC
S184
PORTABLE DEVICE WILL ACQUIRE ENCRYPTED ENCRYPT KEY OF SELECTED CONTENT, DECRYPT WITH SAVE KEY, ENCRYPT WITH COMMUNICATE KEY, AND TRANSFER TO PC
S185
PC WILL DECRYPT TRANSFERRED ENCRYPT KEY WITH COMMUNICATE KEY, AND ENCRYPT WITH ITS OWN SAVE KEY
S186
REGISTER FILE NAME OF SAVED FILE, CONTENT INFORMATION, USER-ENTERED MUSIC PIECE TITLE, ENCRYPTED ENCRYPT KEY, ETC. INTO MUSIC DATA BASE
S187
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S188
END

[FIG. 18]


START COPY FROM EMD SERVER TO HARD DISC
ACCESS FROM PC TO EMD SERVER TO ACQUIRE MUSIC PIECE NO. AND TITLE AND PRICE
S201
DESIGNATE CONTENT TO BE COPIED
S202
MAKE MUTUAL AUTHENTICATION BETWEEN PC AND EMD SERVER AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S203
ACCOUNTING INTERACTION BETWEEN EMD SERVER AND PC
S204
TRANSFER FROM EMD SERVER TO PC DESIGNATED ENCRYPTED MUSIC DATA AND TIME DATA
S205
PC WILL GIVE FILE NAME TO TRANSFERRED MUSIC DATA AND TRANSFER TO HARD DISC
S206
EMD SERVER WILL ENCRYPT ENCRYPT KEY OF THAT CONTENT WITH COMMUNICATE KEY AND TRANSFER TO PC
S207
PC WILL DECRYPT TRANSFERRED ENCRYPT KEY WITH COMMUNICATE KEY, AND ENCRYPT WITH ITS OWN SAVE KEY
S208
MAKE A SET OF FILE NAME OF THAT CONTENT, CONTENT INFORMATION, USER-ENTERED MUSIC PIECE TITLE AND ENCRYPTED ENCRYPT KEY, AND REGISTER INTO MUSIC DATA BASE
S209
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S210
UPDATE RTC BASED ON TIME DATA
S211
END

[FIG. 19]


START ACCOUNTING OPERATION
PC WILL WRITE PRICE INFORMATION OF SELECTED CONTENT INTO ACCOUNTING LOG
S221
PC WILL SEND ACCOUNTING LOG TO EMD SERVER
S222
EMD SERVER WILL COMPUTE FOR ACCOUNTING
S223
S224
ACCOUNTING HAS TO BE SETTLED ?
NO
YES
S225
EMD SERVER WILL SEND TRADE NAME, MONEY AMOUNT, ETC. NECESSARY FOR THAT SETTLEMENT TO SETTLING SERVER
S226
SETTLING SERVER WILL MAKE SETTLEMENT TO USER
RETURN

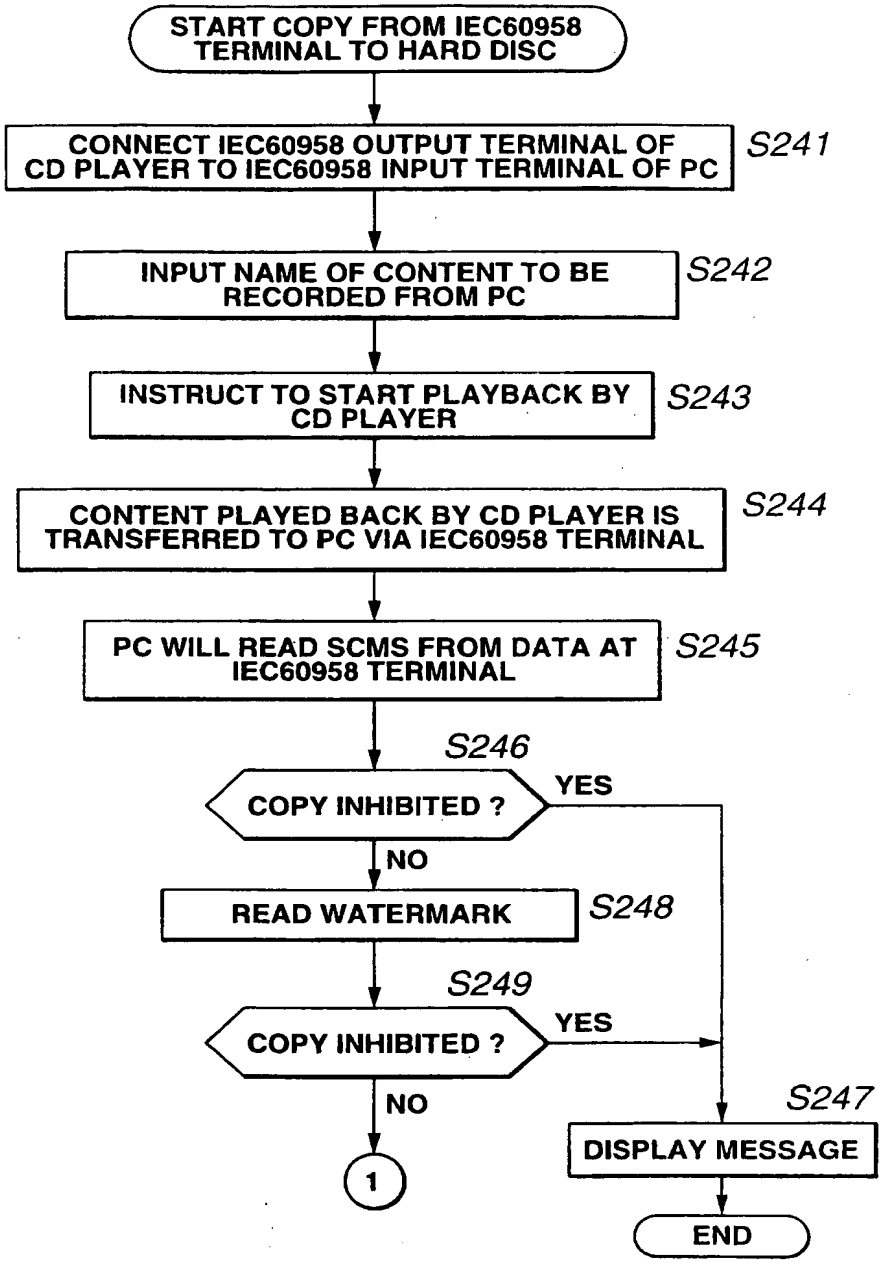
[FIG. 20]

ACCOUNTING LOG

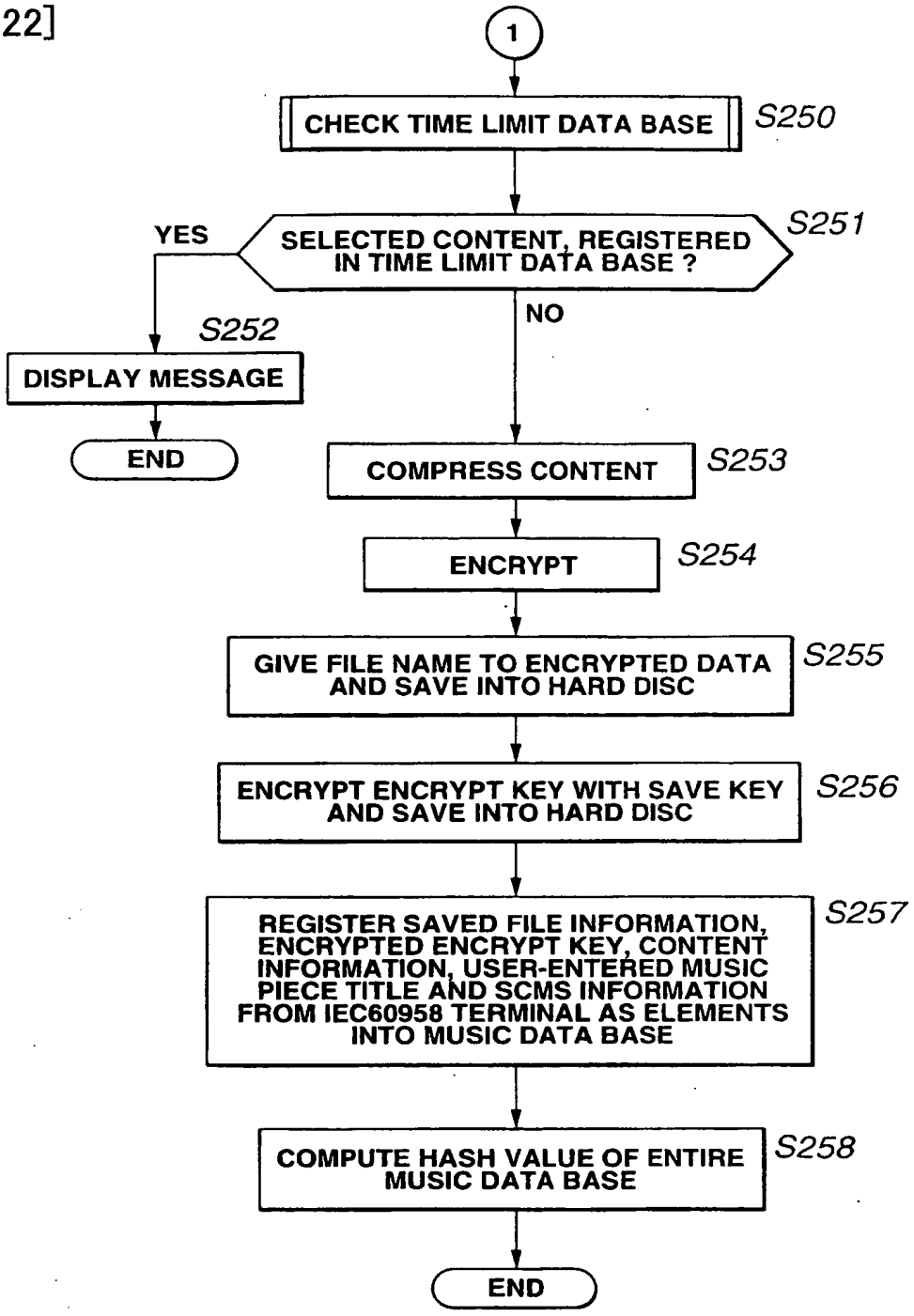
	ITEM 1	ITEM 2	ITEM 3	
FEE	50	50	60	

HASH VALUE	0xf8783e263517

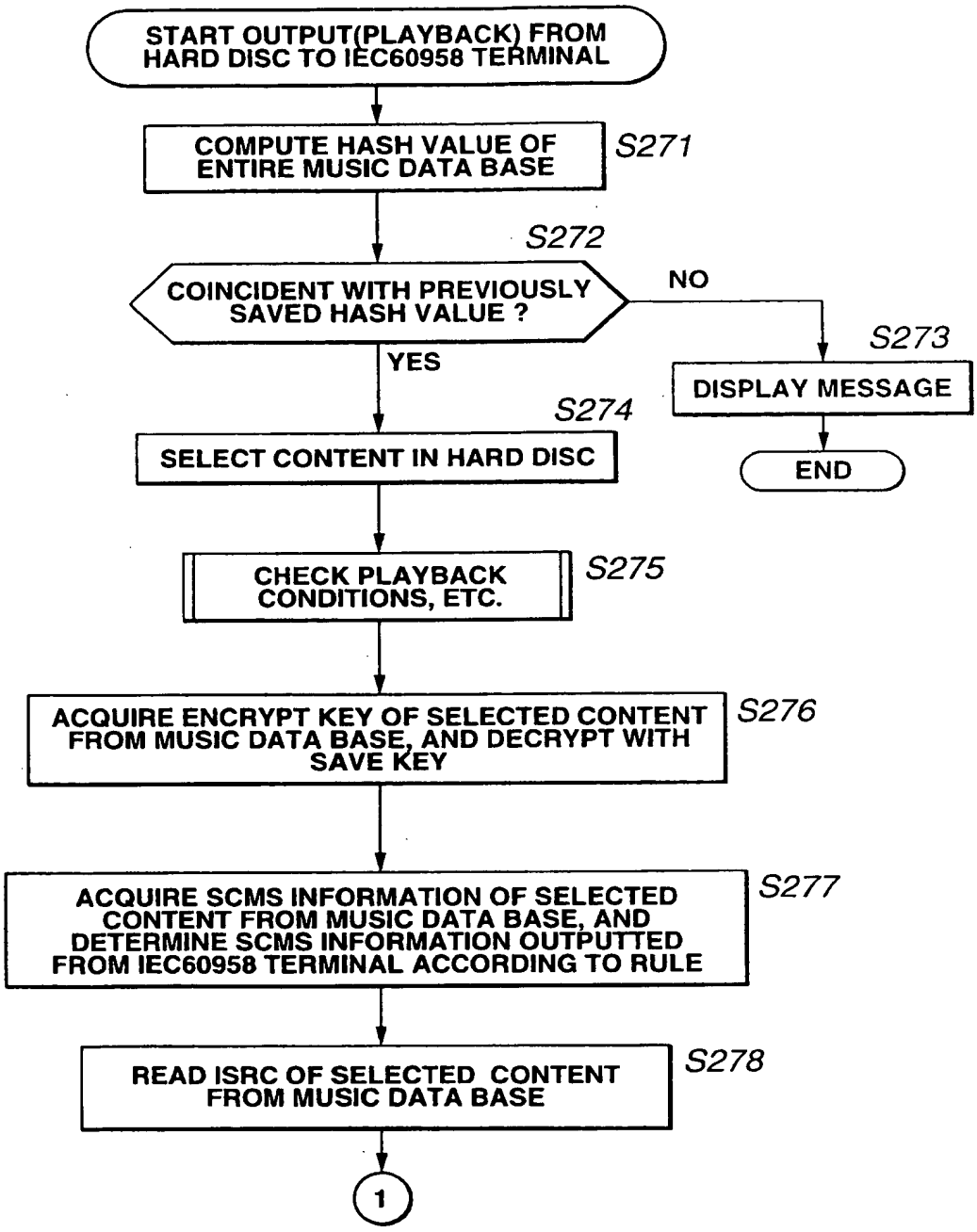
[FIG. 21]


START COPY FROM IEC60958 TERMINAL TO HARD DISC
CONNECT IEC60958 OUTPUT TERMINAL OF CD PLAYER TO IEC60958 INPUT TERMINAL OF PC
S241
INPUT NAME OF CONTENT TO BE RECORDED FROM PC
S242
INSTRUCT TO START PLAYBACK BY CD PLAYER
S243
CONTENT PLAYED BACK BY CD PLAYER IS TRANSFERRED TO PC VIA IEC60958 TERMINAL
S244
PC WILL READ SCMS FROM DATA AT IEC60958 TERMINAL
S245
S246
COPY INHIBITED ?
YES
NO
READ WATERMARK
S248
S249
COPY INHIBITED ?
YES
NO
1
S247
DISPLAY MESSAGE
END

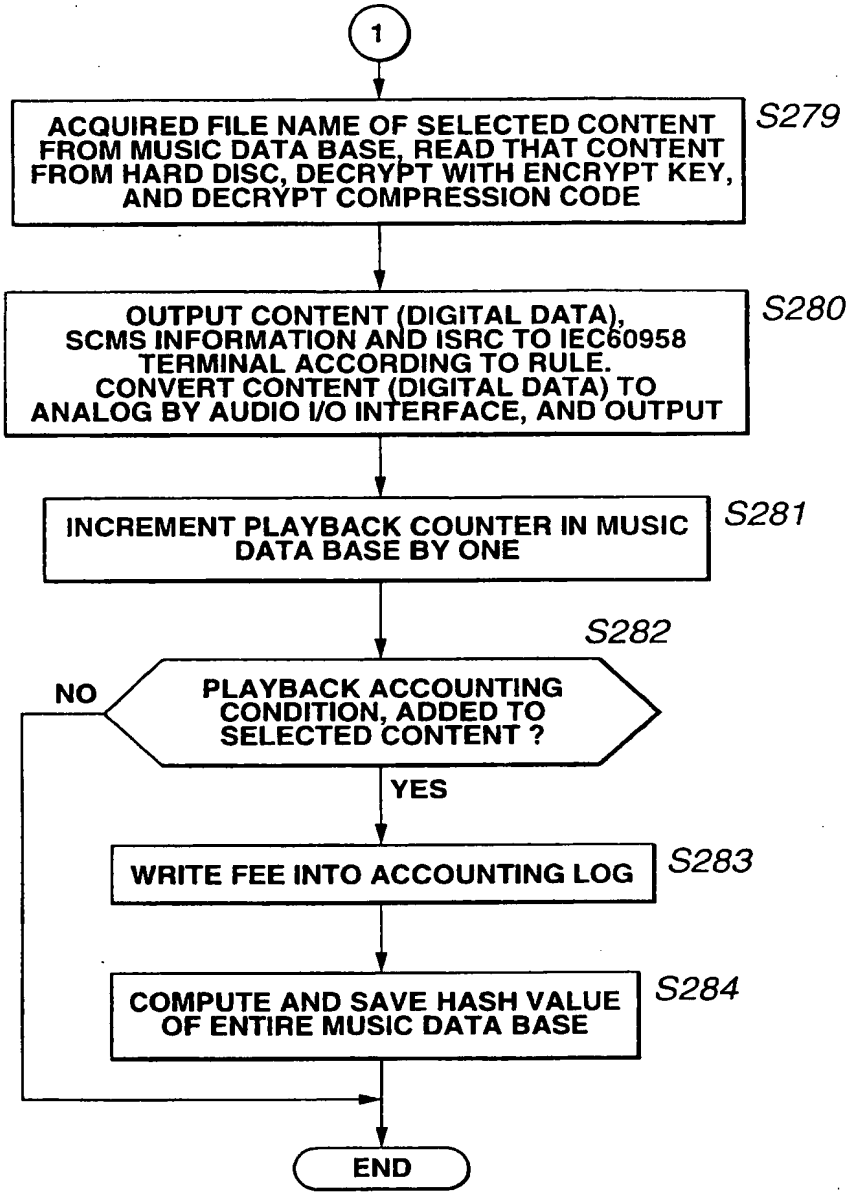
[FIG. 22]


1
CHECK TIME LIMIT DATA BASE
S250
SELECTED CONTENT, REGISTERED IN TIME LIMIT DATA BASE ?
S251
YES
NO
S252
DISPLAY MESSAGE
END
COMPRESS CONTENT
S253
ENCRYPT
S254
GIVE FILE NAME TO ENCRYPTED DATA AND SAVE INTO HARD DISC
S255
ENCRYPT ENCRYPT KEY WITH SAVE KEY AND SAVE INTO HARD DISC
S256
REGISTER SAVED FILE INFORMATION, ENCRYPTED ENCRYPT KEY, CONTENT INFORMATION, USER-ENTERED MUSIC PIECE TITLE AND SCMS INFORMATION FROM IEC60958 TERMINAL AS ELEMENTS INTO MUSIC DATA BASE
S257
COMPUTE HASH VALUE OF ENTIRE MUSIC DATA BASE
S258
END

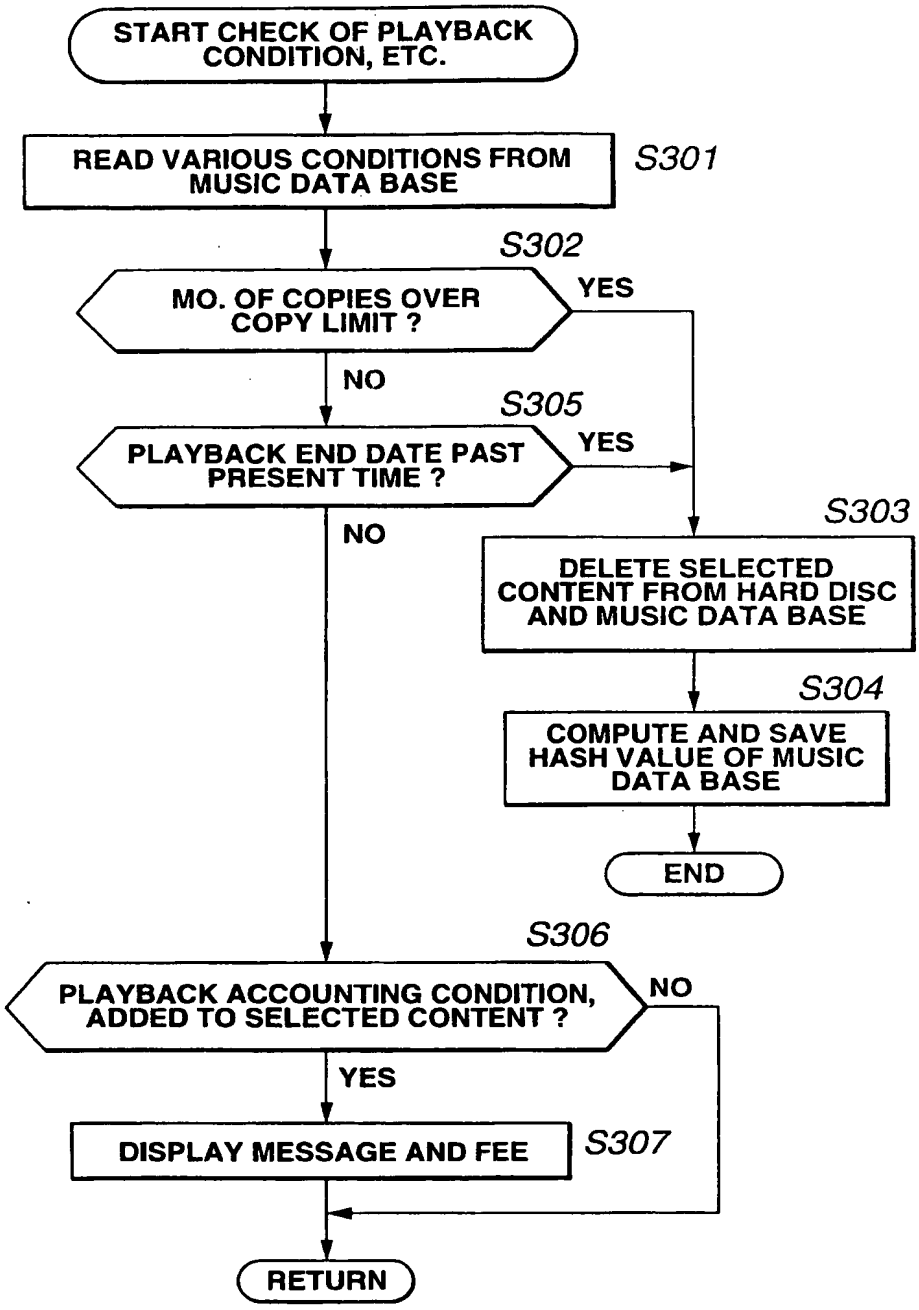
[FIG. 23]


START OUTPUT(PLAYBACK) FROM HARD DISC TO IEC60958 TERMINAL
COMPUTE HASH VALUE OF ENTIRE MUSIC DATA BASE
S271
S272
COINCIDENT WITH PREVIOUSLY SAVED HASH VALUE ?
NO
S273
DISPLAY MESSAGE
END
YES
S274
SELECT CONTENT IN HARD DISC
CHECK PLAYBACK CONDITIONS, ETC.
S275
ACQUIRE ENCRYPT KEY OF SELECTED CONTENT FROM MUSIC DATA BASE, AND DECRYPT WITH SAVE KEY
S276
ACQUIRE SCMS INFORMATION OF SELECTED CONTENT FROM MUSIC DATA BASE, AND DETERMINE SCMS INFORMATION OUTPUTTED FROM IEC60958 TERMINAL ACCORDING TO RULE
S277
READ ISRC OF SELECTED CONTENT FROM MUSIC DATA BASE
S278
1

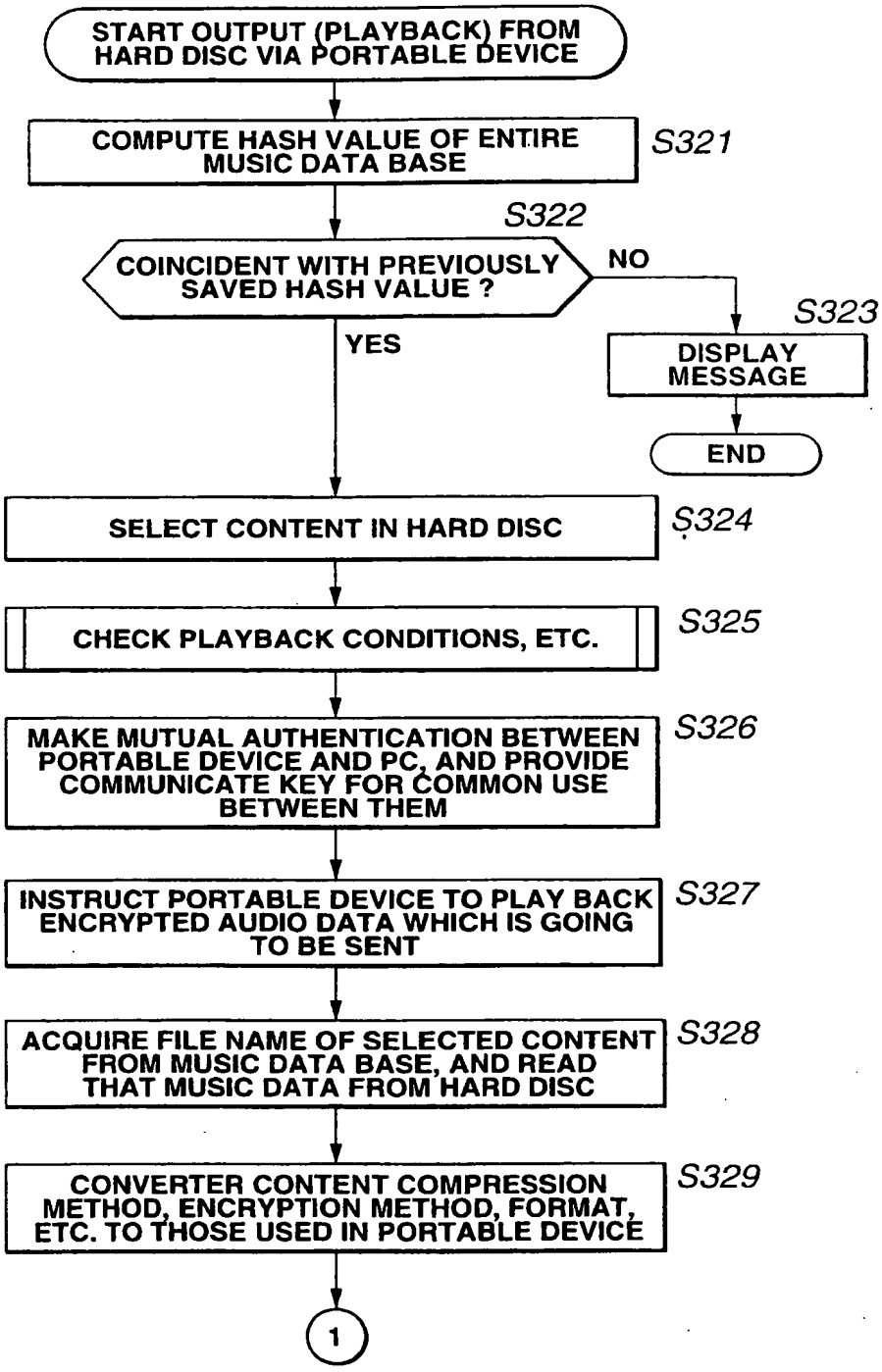
[FIG. 24]


1
ACQUIRED FILE NAME OF SELECTED CONTENT FROM MUSIC DATA BASE, READ THAT CONTENT FROM HARD DISC, DECRYPT WITH ENCRYPT KEY, AND DECRYPT COMPRESSION CODE
S279
OUTPUT CONTENT (DIGITAL DATA), SCMS INFORMATION AND ISRC TO IEC60958 TERMINAL ACCORDING TO RULE. CONVERT CONTENT (DIGITAL DATA) TO ANALOG BY AUDIO I/O INTERFACE, AND OUTPUT
S280
INCREMENT PLAYBACK COUNTER IN MUSIC DATA BASE BY ONE
S281
S282
NO
PLAYBACK ACCOUNTING CONDITION, ADDED TO SELECTED CONTENT ?
YES
WRITE FEE INTO ACCOUNTING LOG
S283
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S284
END

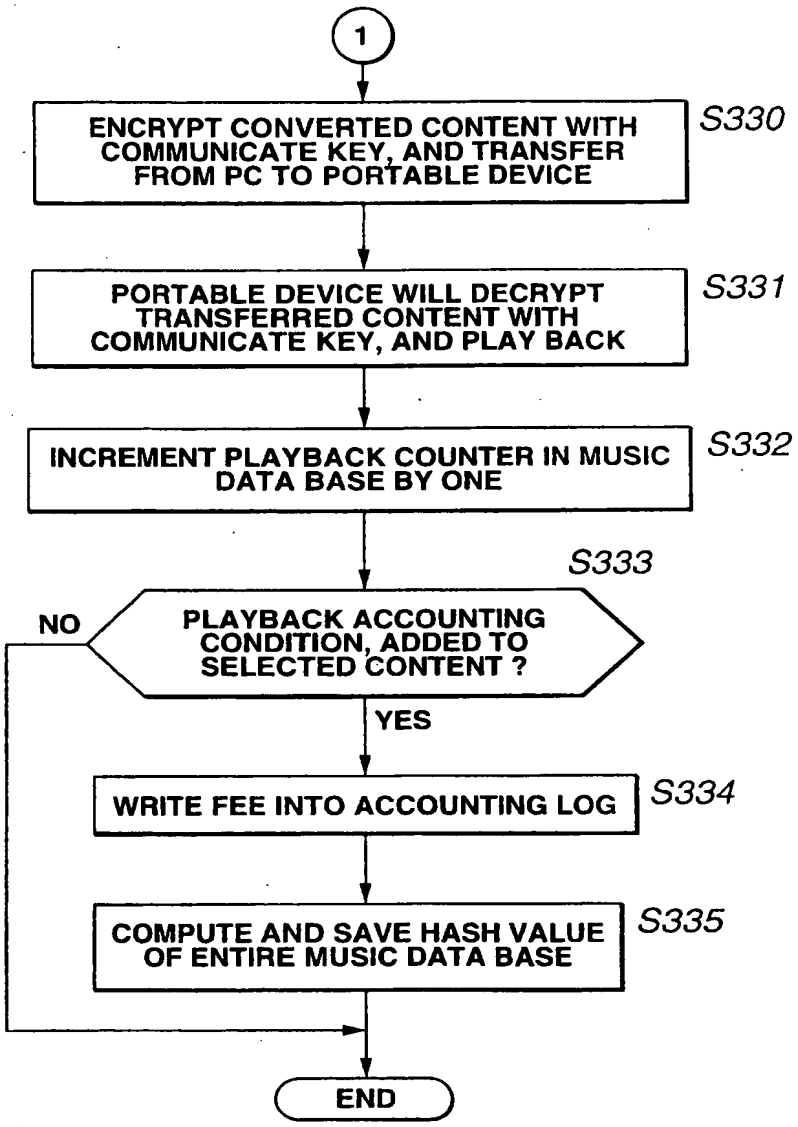
[FIG. 25]


START CHECK OF PLAYBACK CONDITION, ETC.
READ VARIOUS CONDITIONS FROM MUSIC DATA BASE
S301
S302
MO. OF COPIES OVER COPY LIMIT ?
YES
NO
S305
PLAYBACK END DATE PAST PRESENT TIME ?
YES
NO
S303
DELETE SELECTED CONTENT FROM HARD DISC AND MUSIC DATA BASE
S304
COMPUTE AND SAVE HASH VALUE OF MUSIC DATA BASE
END
S306
PLAYBACK ACCOUNTING CONDITION, ADDED TO SELECTED CONTENT ?
NO
YES
DISPLAY MESSAGE AND FEE
S307
RETURN

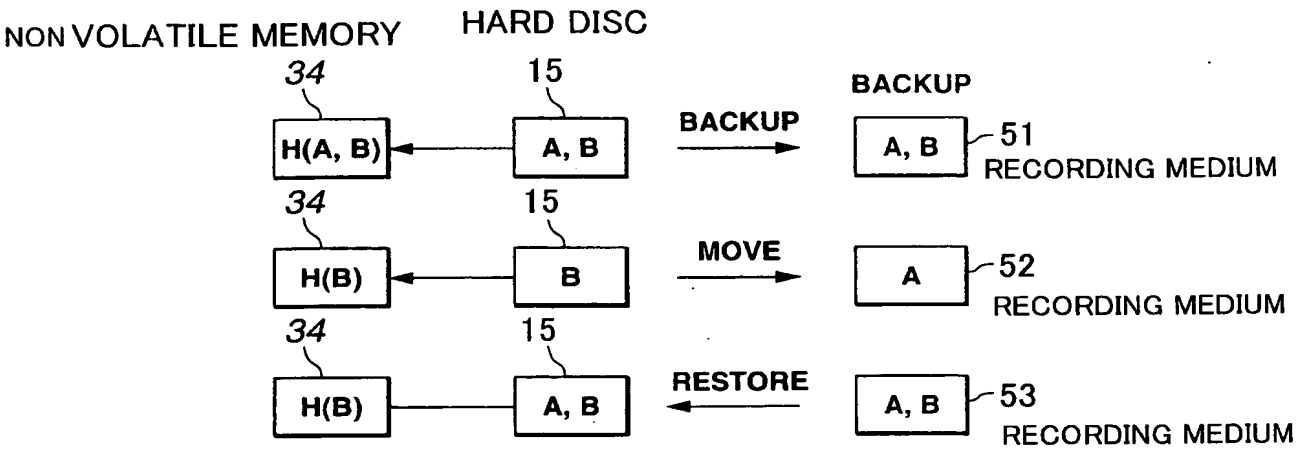
[FIG. 26]


START OUTPUT (PLAYBACK) FROM HARD DISC VIA PORTABLE DEVICE
COMPUTE HASH VALUE OF ENTIRE MUSIC DATA BASE
S321
S322
COINCIDENT WITH PREVIOUSLY SAVED HASH VALUE ?
NO
S323
YES
DISPLAY MESSAGE
END
SELECT CONTENT IN HARD DISC
S324
CHECK PLAYBACK CONDITIONS, ETC.
S325
MAKE MUTUAL AUTHENTICATION BETWEEN PORTABLE DEVICE AND PC, AND PROVIDE COMMUNICATE KEY FOR COMMON USE BETWEEN THEM
S326
INSTRUCT PORTABLE DEVICE TO PLAY BACK ENCRYPTED AUDIO DATA WHICH IS GOING TO BE SENT
S327
ACQUIRE FILE NAME OF SELECTED CONTENT FROM MUSIC DATA BASE, AND READ THAT MUSIC DATA FROM HARD DISC
S328
CONVERTER CONTENT COMPRESSION METHOD, ENCRYPTION METHOD, FORMAT, ETC. TO THOSE USED IN PORTABLE DEVICE
S329
1

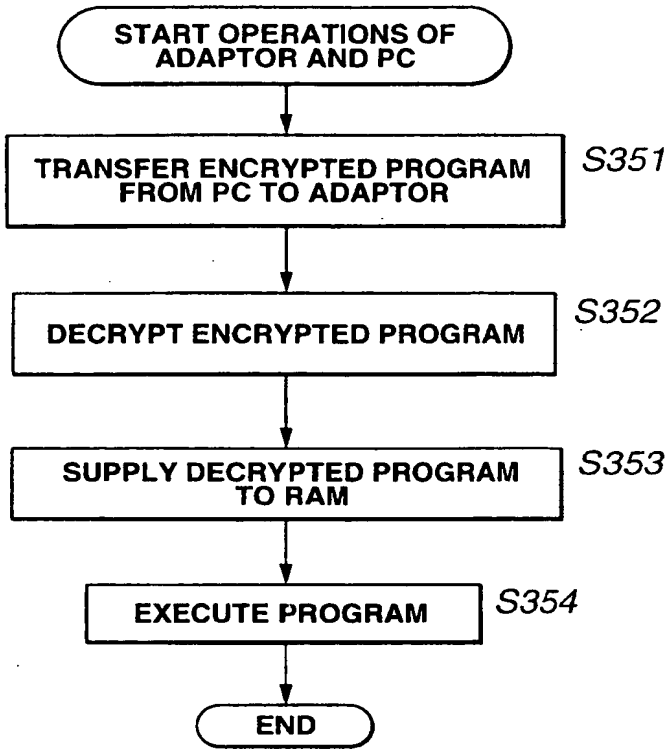
[FIG. 27]


1
ENCRYPT CONVERTED CONTENT WITH COMMUNICATE KEY, AND TRANSFER FROM PC TO PORTABLE DEVICE
S330
PORTABLE DEVICE WILL DECRYPT TRANSFERRED CONTENT WITH COMMUNICATE KEY, AND PLAY BACK
S331
INCREMENT PLAYBACK COUNTER IN MUSIC DATA BASE BY ONE
S332
S333
NO
PLAYBACK ACCOUNTING CONDITION, ADDED TO SELECTED CONTENT ?
YES
WRITE FEE INTO ACCOUNTING LOG
S334
COMPUTE AND SAVE HASH VALUE OF ENTIRE MUSIC DATA BASE
S335
END

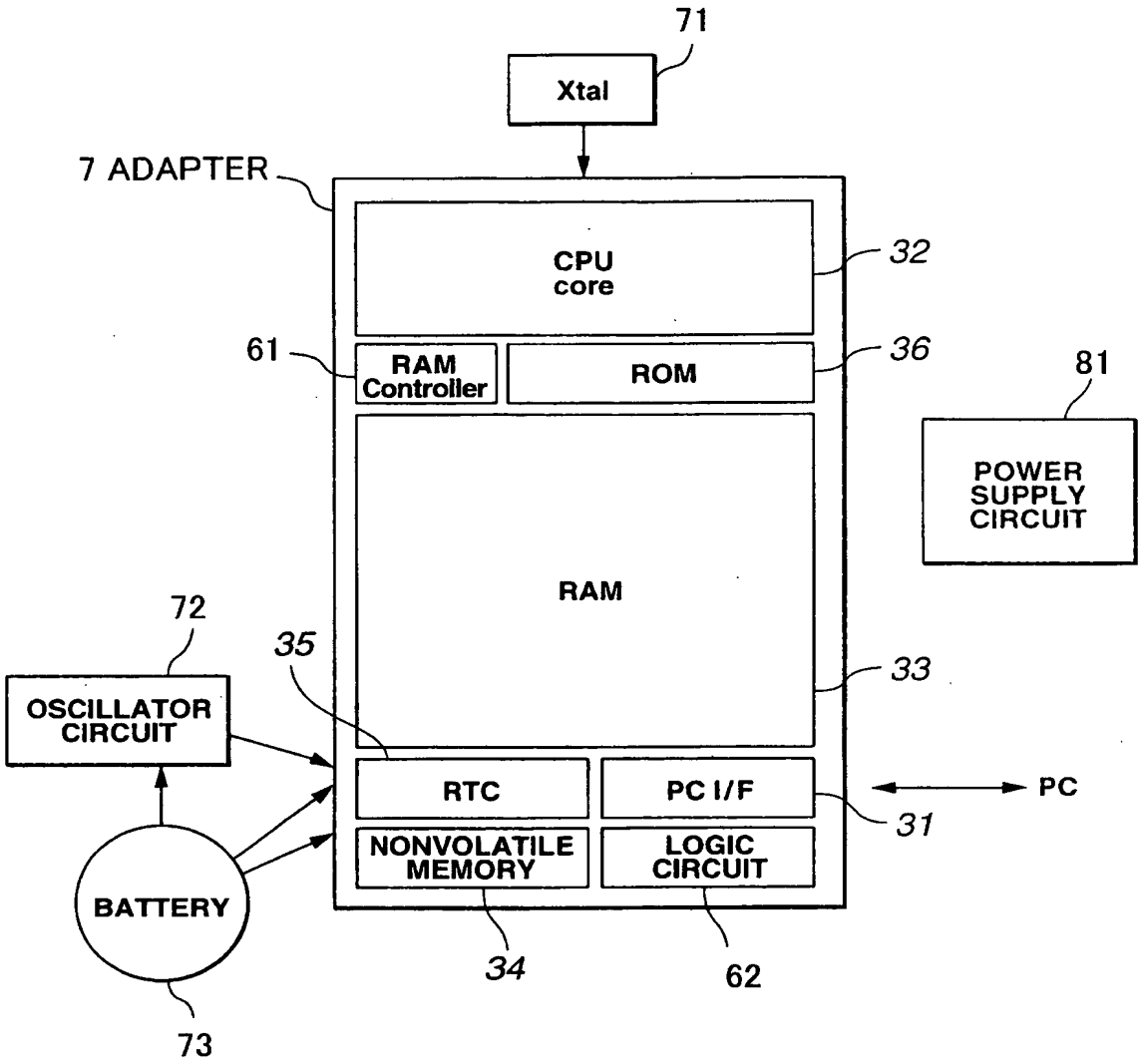
[FIG. 28]


NON VOLATILE MEMORY
HARD DISC
34
15
H(A, B)
A, B
BACKUP
BACKUP
A, B
51
RECORDING MEDIUM
34
15
H(B)
B
MOVE
A
52
RECORDING MEDIUM
34
15
H(B)
A, B
RESTORE
A, B
53
RECORDING MEDIUM

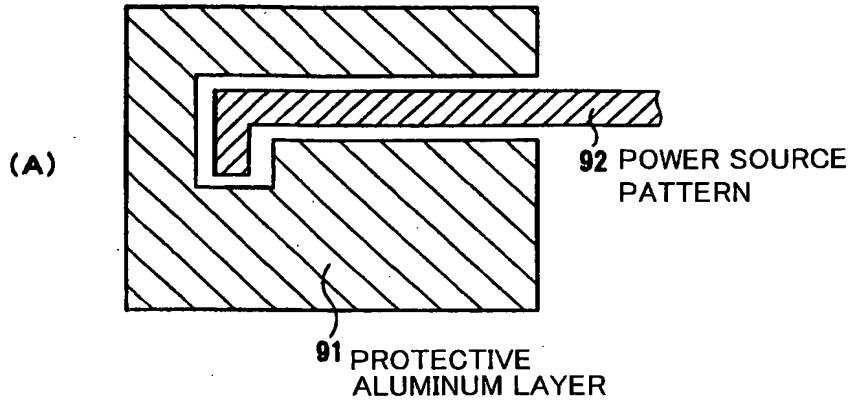
[FIG. 29]

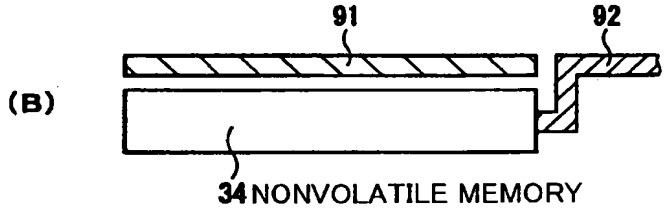

START OPERATIONS OF ADAPTOR AND PC
TRANSFER ENCRYPTED PROGRAM FROM PC TO ADAPTOR
S351
DECRYPT ENCRYPTED PROGRAM
S352
SUPPLY DECRYPTED PROGRAM TO RAM
S353
EXECUTE PROGRAM
S354
END

[FIG. 30]

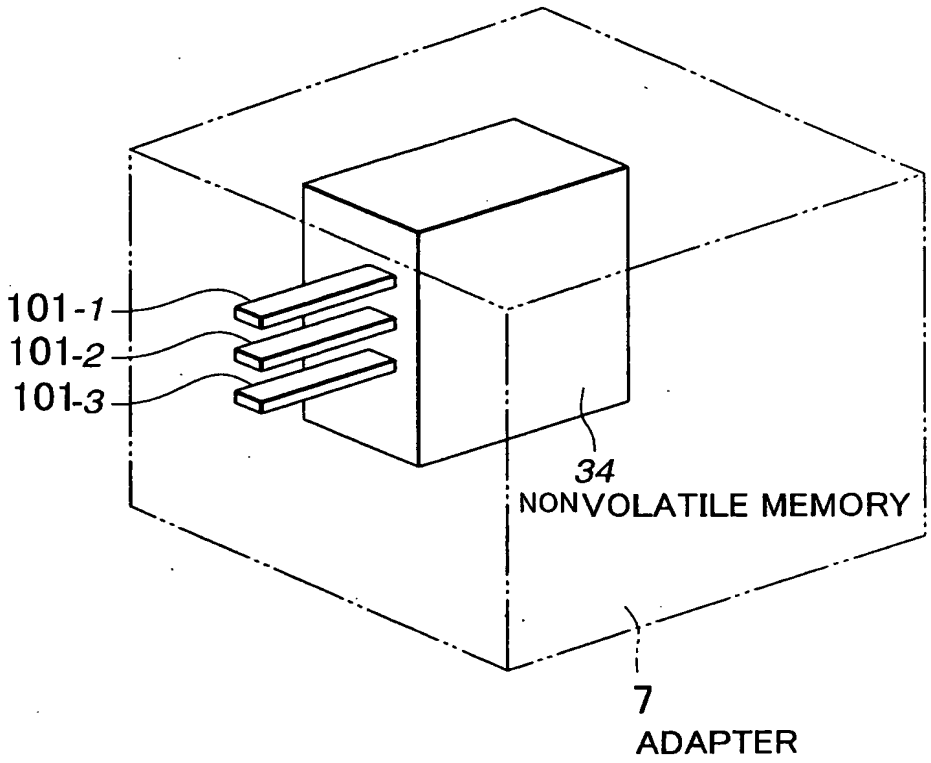

71
Xtal
7 ADAPTER
CPU core
32
61
RAM Controller
ROM
36
81
POWER SUPPLY CIRCUIT
RAM
72
35
33
OSCILLATOR CIRCUIT
RTC
PC I/F
PC
31
NONVOLATILE MEMORY
LOGIC CIRCUIT
BATTERY
34
62
73

[FIG. 31]


(A)
92 POWER SOURCE PATTERN
91 PROTECTIVE ALUMINUM LAYER


(B)
91
92
34 NONVOLATILE MEMORY

[FIG.32]



[Name of Document] ABSTRACT

[Summary]

[Task]

To prevent falsification of a software used with data in order to inhibit fraudulent copying of the data.

[Means for Solution]

A CPU 12 of a personal computer 1 controls a CPU 32 of an adapter 7 comprised of a semiconductor IC to calculate hash values of a terminal database which manages music data recorded in a hard disc 15 and to store it in a nonvolatile memory 34. When the music data recorded in the hard disc 15 is reproduced, the CPU 12 calculates the hash values of the terminal database in the hard disc 15, compares it with the original hash value stored in the nonvolatile memory 34, and controls the reproduction of the music data from the hard disc 15 corresponding to the result of the comparison.

[Selected Drawing] Fig.1